REALISTIC MOBILITY MODELING AND SIMULATION FOR MOBILE WIRELESS NETWORK IN URBAN ENVIRONMENTS

by

Jonghyun Kim

A thesis submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Master of Electrical Engineering in Electrical and Computer Engineering

Summer 2005

REALISTIC MOBILITY MODELING AND SIMULATION FOR MOBILE WIRELESS NETWORK IN URBAN ENVIRONMENTS

by

Jonghyun Kim

Approved: ____
Stephan K. Bohacek, Ph.D.
Professor in charge of thesis on behalf of the Advisory Committee

Approved: ____
Gonzalo R. Arce, Ph.D.
Chair of the Department of Electrical and Computer Engineering

Approved: ____
Eric W. Kaler, Ph.D.
Dean of the College of Engineering

Approved: ____
Conrado M. Gempesaw II, Ph.D.
Vice Provost for Academic and International Programs

ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to Dr. Stephan Bohacek for guidance, encouragement, and many hours that he spent in my behalf. What I remember strongly is that whenever I confronted very hard problem with research and I had very low volition, he told me "Please don't give up". These four words gave me great courage. He is a great mentor.

I also express my appreciation to Dr. MyungSuk Jo, Dr. SangMin Lee, and Dr. BoHyeun Wang who gave me great opportunity to study abroad and have been guiding me. Without their help, I am not even able to say this appreciation.

I want to thank Vinay Sridhara, Andjela Ilic, and Khushboo Shah. They have helped me a lot about my research and have improved my language skills. It is a pleasure to work with them. I will not forget them.

Finally, I am very grateful to my family, OkBun Kim and SeungHyun Kim. They always give me full attention with positive attitude and provide me with unconditional support of every kind.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Mobility clearly is an integral issue in mobile wireless networks. However, little research has focused on realistic mobility. Unrealistic mobility models, such as the popular Random Waypoint or Random Walk models, may result in incorrect conclusions regarding the performance evaluation of mobile wireless network protocols.

This thesis introduces a mobility model and simulator that attempt to generate realistic mobility for mobile wireless network simulation in an urban setting. The mobility models are based on data collected in a number of surveys. The result is a 3-layer hierarchical model. The highest layer is an activity model that determines the high level activity (e.g., working). The second layer is a task model that determines the specific task within an activity (e.g., meeting with three people). The third layer is an agent model that determines how mobile nodes move from one location to another (e.g., how a node navigates down a crowded hallway).

A mobility simulator implements the mobility model and is based on the technique of discrete event simulation with rescheduling. This simulator is the first realistic simulator for mobile wireless networks. It is hoped that this simulator will greatly increase the fidelity of mobile wireless network performance evaluation. Further, it is hoped that this work is usher in a new investigation of urban mobile wireless networks.

Chapter 1

INTRODUCTION

1.1 Introduction

Mobile ad hoc networks (MANETs) are networks of mobile nodes (MNs) that happen to be near each other with no pre-established infrastructure, forming a temporary network [1]. Each MN takes part in the process of forwarding packets (i.e. each MN is both a router and a host). A packet generated by a MN may reach a destination node through multiple MNs, that is the packet travels along a multi-hop path. At a given time, a MANET graph (where the vertices are the nodes and arcs are the between nodes that are able to communicate) depends on the MNs' positions and their transmitter and receiver coverage patterns, which depends on the transmission power levels and the location of obstacles (e.g. buildings and trees) as well as on the physical layer used and receiver sensitivity, etc. [2]. The greatest challenge of MANETs is that this multi-hop graph changes as the MNs move.

In spite of this major challenge, researchers have paid tremendous attention to mobile wireless ad hoc networks. First, MANETs do not require any infrastructure. Thus, they can be quickly deployed. Second, ad hoc networks are self-organizing and self-healing. If a new node is inserted into the network, the network discovers the new node and automatically incorporates it into the network without the need for a system administrator. Likewise, if some nodes fail or leave the network, the network also automatically soon senses this occurrence and re-figures the multi-hop graph. As a result, ad hoc networks provide high reliability and adaptability.

Wireless systems for industry have mostly used cellular-phone-style radio links, using point-to-point or point-to-multipoint transmission [3]. Note that cellular networks are single-hop networks and MANETs are multi-hop networks. A point-to-point network can provide reliability because there are only two nodes, but it does not scale to handle more than one pair of nodes. A point-to-multipoint network can handle more nodes, but its reliability is determined by the placement of the access point and mobile nodes. Moving an access point (AP) to improve communications with one mobile node will often degrade communications with other mobile nodes. However, unlike single-hop networks, in multi-hop networks, even if a node cannot connect to the AP directly because of environmental constraints (e.g. obstacles), it will find another route to AP through other MNs. Thus, multi-hop networks are more reliable and adaptable to environmental constraints than are single-hop networks. Multi-hop networks are also scalable to thousands of mobile nodes due to distribution algorithm. A third reason is that ad hoc networks can provide higher bandwidth, and spatial reuse. The communication ranges used in MANETs are short. The physics of wireless communication dictates that the bandwidth be higher at shorter range because of low interference and path loss. Additionally, little power is required to transmit data across multiple short distances. Since the shorter transmission ranges limit interference, simultaneous and spatially separated data flows are allowed.

An example of MANETs is an ad-hoc network of laptops and PDAs communicating with each other over a wireless medium. Another example is an ad-hoc network of emergency responders and ambulances in disaster recovery situations and places with non-existing or damaged communication infrastructure where rapid deployment of communication network is needed. Third example is an ad-hoc network of soldiers and military vehicles in a battlefield.

Planning has been done to deploy a citywide ad hoc networks (i.e. urban mesh networks) for civilian applications. The city of Philadelphia is in the final planning stages of providing wireless access to the entire 135 square mile city [4]. Atlanta, Los Angeles, Milwaukee and Cleveland have each advanced plans for developing their own wireless network and more than 200 local governments are in the early stages of plotting ways to provide residents with cheap wireless broadband. MNs (i.e. mobile devices having wireless equipment) may be located in or on people, cars, trucks, airplanes, ships, etc. Urban mesh networks usually consist of fixed base stations, fixed wireless relays and mobile nodes. A fixed base station is the interface between wired network and wireless network, that is, it is the path to Internet from the MNs. In a city, those three elements eventually make a dynamic multi-hop graph.

No one has tried to simulate realistic dynamic multi-hop graph in urban environments. Random Waypoint and Random Walk, while popular for modeling these graphs, do not provide realistic models of urban civilian mobility. The reason is that realistic mobility must account for various structures that arise in urban civilian mobility. Specifically, pedestrians move along sidewalks and inside buildings, while vehicles move along roads. Furthermore, pedestrians and vehicles obey various rules. For example, vehicles and pedestrians, more or less, obey traffic signals. Also, in order for a faster node to pass a slower node, it must go around the slower node. If there is no room to pass, then the faster node must decrease its speed to that of the slower node and follow until there is room to pass. It should be noted that only if mobility models are realistic, can we get realistic dynamic multi-hop topology and hence be assured of the correctness of conclusions draw from simulations. One way to model realistic civilian mobility is to do rigorous surveys of extensive pedestrian and vehicle literatures to get proven distributions (e.g., people's walking speed, vehicle's driving speed, list of activities, etc.) to be used in simulation.

Simulation techniques such as discrete event simulation (DES) and continuous simulation have been developed to imitate real world systems. Within a DES

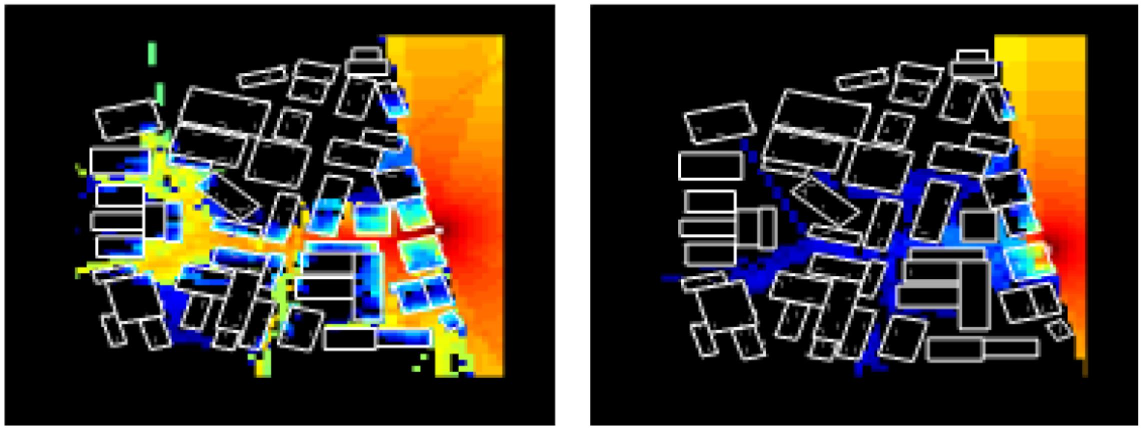

Figure 1.1: Signal Strength in an Urban Environment. The colors indicate the signal strength from a transmitter on the right-hand side of the modeled area. While the figures show a large variation in coverage, the only difference between the two figures is that the position of the transmitter has moved 10 meters [6].

mechanism, time is controlled by discrete time at which predefined events occur, while within a continuous simulation mechanism, time is controlled by continuous variables expressed as differential equations that depend on a system [5]. Validation is one of those which should be checked prior to experimental analysis. Validation means that the simulator behaves with satisfactory accuracy consistent with the real world system.

This thesis has made an effort to derive realistic mobility models for urban civilian through rigorous surveys, implement mobility simulator based on DES mechanism, and to validate designed mobility simulator with real measured results. It is hoped that the UDel mobility simulator leads to accurate protocols performance evaluation and initiates a more thorough examination of wireless networks in urban areas.

1.2 Motivation

Figure 1.1 shows that the small movement of the MN significantly affects path loss [6]. Clearly, the signal strength of every point from source node in the three dimensional space quickly varies due to node mobility. This means that node mobility determines properties of the topology and the behavior of the links among MNs. It stands to reason that the performance of MANETs is also strongly impacted by mobility and propagation. However, such a hypothesis cannot be checked until a mobility model is developed, which is the purpose of this thesis.

1.3 Thesis Overview

The remainder of this thesis presents related research areas in Chapter 2. The overview of projects that our group has worked on is briefly introduced in Chapter 3. Details of mobility models are described in Chapter 4. Chapter 5 explains how to design mobility simulator. Simulation environment and the results are shown in Chapter 6 and Chapter 7, respectively. Chapter 8 concludes with ideas for future research.

Chapter 2

RELATED WORK

Modeling node movement to reflect the real world presents major challenges. The best way to model node movement is to observe real user-participants and trace their movements over an adequate time period. Since this direct experimentation is expensive and time consuming, researchers have attempted through simulation to realistically represent the behaviors of MNs without tracing real ones. Their efforts have supplied a variety of mobility models. It is important to note, that mobility modeling is widely used outside of MANET performance evaluation. Here we discuss mobility models used in MANET performance evaluation, as well as models used in other areas.

2.1 Non-graph-based Mobility Models

Non-graph-based Mobility Models are the mobility models that allow nodes to move freely in which there are none of the obstacles that exist in a real city. Popular mobility models within this class of models includes Random Waypoint, Random Walk, Random Direction, Boundless Simulation Area, Gauss-Markov, and Probabilistic Version of Random Walk Mobility Models [7].

By far, the most popular of all mobility models used for MANET performance evaluation is the Random Waypoint Mobility Model. The Random Waypoint Model uses a MN which pauses for a specified period and then starts traveling by randomly choosing a random destination and speed. Upon reaching the destination, the node waits for a uniformly distributed pause time. After waiting, the node chooses another destination and speed, continuing the process. By introducing pause times, this model attempts to solve the problem of sudden stops and sharp turns that the Random Walk model has. However, the model causes density waves, which means the clustering of nodes near the center of the simulation area.

In the Random Walk Mobility Model a MN starts traveling by randomly choosing a direction and speed. After either a specified time or a specified distance, the node chooses a new direction and speed. If the node reaches a boundary of the simulated area, it bounces off the simulation border at an angle determined by the incoming direction. The node then continues along this new path. This model can generate unrealistic movements such as sudden stops and sharp turns.

Random Direction Mobility Model was developed to overcome density waves. In this model, a MN chooses a direction and a speed, and then travels in that direction until the node arrives at the border of the simulation area. Upon reaching the boundary, it pauses for a specified time, then chooses another direction and speed. Since this model makes for uniform node density and since nodes pause only at the border of the modeled area, it is not realistic.

Boundless Simulation Area Mobility Model aims to allow a MN to travel unobstructed in a simulation area by handling the boundary of the simulation area differently, thus removing any simulation edge effects from the performance evaluation. The key idea is that the node that reaches one side of the simulation area continues traveling and reappears on the opposite side of the simulation area. But, the problem is that one static node and one MN that continues to move in the same direction become neighbors repeatedly. In addition, the channel model should be modified to wrap transmissions from one edge of the area to the other.

Gauss-Markow Mobility Model was designed to adapt to different levels of randomness via one tuning parameter. Each MN is assigned an initial current speed and direction. The next speed and direction of a node is calculated based on the current speed, direction, and a random variable, whereas the next position is calculated based on the current position, speed, and direction. The following equations represent this model:

$$\begin{aligned} s_n &= \alpha s_{n-1} + (1-\alpha)\overline{s} + \sqrt{(1-\alpha^2)} s_{x_{t-1}} \\ d_n &= \alpha d_{n-1} + (1-\alpha)\overline{d} + \sqrt{(1-\alpha^2)} d_{x_{t-1}} \\ x_n &= x_{n-1} + s_{n-1} \cos d_{n-1} \\ y_n &= y_{n-1} + s_{n-1} \sin d_{n-1} \end{aligned}$$

where s_n and d_n are the next speed and direction at time interval n; x_n and y_n are the next x and y coordinate of the node's position; $\overline{s}$ and $\overline{d}$ are constants representing the mean value of speed and direction as $n \longrightarrow \infty$; $s_{x_{t-1}}$ and $d_{x_{t-1}}$ are random variables from a Gaussian distribution; α, where $0 \leq \alpha \leq 1$, is the tuning parameter. If $\alpha = 0$, s_n, d_n, x_n, and y_n become totally random values (i.e. Brownian motion). If $\alpha = 1$, since $s_n = s_{n-1}$, and $d_n = d_{n-1}$, linear motion is obtained. Thus, the model can adjust the level of randomness via α.

Finally, Probabilistic Version of Random Walk Mobility Model utilizes a probability matrix to determine the position of a MN in the next time step. There are three states used in the matrix. State 0 represents the current (x or y) position, state 1 represents the previous position, and state 2 represents the next position if the node continues to move in the same direction. The probability matrix is:

$$\begin{bmatrix} P(0,0) & P(0,1) & P(0,2) \\ P(1,0) & P(1,1) & P(1,2) \\ P(2,0) & P(2,1) & P(2,2) \end{bmatrix}$$

where each entry $P(a,b)$ represents the probability that an MN will go from one state a to state b. The main advantage is that we can produce probabilistic movements of MNs by adjusting the probability matrix. However, choosing appropriate values of $P(a,b)$ may prove difficult.

None of the mobility models mentioned above are suitable for simulation of urban mesh networks because first, the models do not even consider real city area, and second, they do not generate realistic node movements in terms of urban environments. If someone uses one of these models in an urban area, it is easy to imagine that people try to pass through walls and cars leave the roads or drive into rivers.

2.2 Graph-based Mobility Models

In this model, MNs can only move along the predefined graph including vertexes and arcs generated based on a real city map. City Section, Freeway, Manhattan, Obstacle, and Graph-based Mobility Models belong to this category [7], [8], [9], [10].

In the City Section Mobility Model, the modeled area is a street network that represents a section of a city. All MNs must follow predefined paths and behavior guidelines (e.g. traffic laws). Without regard to obstacles and traffic regulations, MNs do not have the ability to roam freely in the real world. Each MN begins the simulation at a defined point on some street, and then randomly chooses a destination which is also predefined by a point on some street. The node travels to a destination through the shortest path between two points. The speed is based on the speed limits on the streets. Upon reaching the destination, the node pauses for a specified time, then chooses another destination and repeats the process. This model, as well as the rest of the graph-based models discussed here, does not account for traffic lights or congestion. That is, nodes are allowed to drive through each other.

In Freeway and Manhattan Mobility Models, freeway and Manhattan maps are used. The freeway map consists of several freeways, and each has lanes in both directions. The Manhattan map consists of a number of horizontal and vertical streets forming a grid, and each street has two lanes in each direction. At an intersection of a horizontal and a vertical street, the node can turn left with the probability of 0.25, right with 0.25, and go straight with 0.5. The Freeway model does not allow a node to change the lane it is driving in, while the Manhattan model gives a node some freedom to change its direction. Like the City Section Model, these models do not account for inter-node and intra-node relationships in both models. However, these models do not enforce node speed limits.

Obstacle Mobility Model allows a user to define the positions of obstacles (e.g. buildings). The movement graph, which is a set of pathways along which the mobile nodes move, is defined by the Voronoi Diagram of the obstacle corners. This model uses the shortest path routing policy to move the nodes between two locations in the movement graph. The algorithm of movements of MNs is based on Random Waypoint Mobility Model. An important distinction between this model and City Section model is that the City Section can consider a real city. The Manhattan model considers a idealized city that has a perfect grid layout. However, the Obstacle Mobility Model places buildings at random locations, hence blocks do not exist and streets are never straight.

In Graph-based Mobility Model, a graph consists of vertices and edges. The vertices represent locations (e.g. theater, city hall, church, etc.) that MNs visit. The edges model the connections between these locations. Because one vertex is just one building, it may scarcely reflect real-world terrains. Each MN is initialized at a random vertex in the graph. The algorithm of movements of MNs is based on Random Waypoint Mobility Model.

Although the predefined graph is more or less based on a real map, all the mobility models mentioned in this section still use Random Waypoint or Random Walk Mobility Model as the algorithm of movements of MNs (with the restriction that nodes must remain on the graph), and hence the models are not realistic.

2.3 GEMM

One of the most detailed mobility models is GEMM [11]. GEMM is an activity-based and agent-based model where several factors impact the mobility of the node. For example, GEMM includes attraction points as well as habits to influence the mobility. A notable drawback of this work is that realistic values of the model parameters are not known. However, our mobility model, which is called UDel mobility model and integrates graph-based, group, activity-based, and agent-based mobility models, is based on proven approaches, and, perhaps more importantly, uses parameters that are verified through observation. Therefore, the user will not be burdened or even justified in arbitrarily setting parameters. Rather, the goal is to develop a few parameter choices such as the map and the time of day (e.g., morning rush hour). Of course to allow exploration, parameter settings can be modified as desired.

2.4 Group Mobility Models

In ad hoc networks, there are many situations where it is necessary to model the behavior of MNs as they move together (e.g. emergency response team, rescue party, a platoon of soldiers, etc.). In this section, Exponential Correlated Random, Column, Nomadic Community, Pursue, and Reference Point Group Mobility Models as group mobility models are explained [7]. Exponential Correlated Random Mobility Model is one of the first group mobility models. In this model, a motion function is used to create MN movements. Given a position at time t, $\overrightarrow{b}(t)$ is used to define the next position at time $t+1$, $\overrightarrow{b}(t+1)$:

$$b(t+1) = b(t)e^{-\frac{1}{\tau}} + (\sigma\sqrt{1-(e^{-\frac{1}{\tau}})^2})r$$

where τ adjusts the rate of change from the node's previous location to its new location and r is a random Gaussian variable with variance σ.

Column Mobility Model represents a set of MNs that move around a given reference grid forming a column of MNs and consisting of reference points for each node, which is moving in a forward direction. An initial reference grid is defined and then each node is placed in relation to its reference point in the reference grid. Each node is allowed to move randomly around its reference point. The new reference point for a given node is defined as:

$$new\_reference\_point = old\_reference\_point + advance\_vector$$

where *advance_vector* is a predefined offset that is computed via a random distance and a random angle. The predefined offset moves the reference grid. An example of this model is a row of soldiers marching together towards the enemy.

Nomadic Community Mobility Model represents groups of MNs that move together from one point to another. In this model, each node roams around a given reference point. When the point of reference changes, all nodes in the group drift to the new area defined by the reference point and then begin roaming around it again. The difference between Column and Nomadic Community Model is that the nodes in Column Models share a common reference point. An example is a class of students touring an art museum.

Pursue Mobility Model represents a group of MNs tracking a certain target. The new position is defined as:

$$new\_position = old\_position + acceleration(target - old\_position) + random\_vector$$

where *acceleration(target - old_position)* is the acceleration function to pursue the target appropriately and *random_vector* is a random offset for each MN. An example of this model is police officers attempting to catch an escaped criminal.

Reference Point Group Mobility (RPGM) model represents group movements via a logical center within a group and group motion vector $\overrightarrow{GM}$. There are reference points whose numbers correspond the number of MNs in a group. A logical center decides the next reference points of group motion by using group motion vector. A new reference point for each MN in the group is then calculated by its old reference point plus $\overrightarrow{GM}$. A new position for each MN is then calculated by summing a random motion vector, $\overrightarrow{RM}$, with the new reference point. $\overrightarrow{GM}$ is predefined according to the objectives of group mobility. The length of $\overrightarrow{RM}$ is uniformly distributed within a specified radius centered at reference point and its direction is uniformly distributed between 0 and 2π. An example of this model is a team consisting of human and dog responding to avalanche rescue.

2.5 Mobility models from other disciplines

Within disciplines such as urban planning, architecture, transportation engineering, and sociology, mobility modeling is a mature field with early efforts dating back nearly fifty years. Much of the previous work in mobility can be classified into flow based [12], [13], [14], [15], [16], mesoscopic [17], [18], cellular automata [19], [20], [21], agent-based [22], [23], [24], [25], [26], [27], and activity-based methods [28], [29], [30], [31]. Flow-based methods do not model individual mobile nodes, but the density of nodes in continuous flows. Mesoscopic models aggregate nodes into groups. Cellular automata cellulates space and model the node density in each cell. Since mobility modeling in MANETs requires each node to be modeled individually, these three methods are not appropriate for our needs. However, agent-based and activity-based methods model each node individually. Agent-based models dictate that the nodes follow simple rules, perhaps with some random decisions and with interactions between other nodes and the surroundings (e.g., nodes avoid colliding with other nodes and buildings). Activity-based models define a specific list or schedule of activities that each node performs (e.g., go to work, go to restaurant). Typically, activity-based models incorporate agent-based models since the tasks are performed by agents. More exactly, the activity model determines a task, and then the agent model performs the task.

Realistic city graph and mobility models are keys to simulate realistic mesh networks in urban environments. To this end, it should be noted that realistic city graph must be drawn from real maps or from GIS information, and likewise, realistic mobility models must be derived from extensive research areas dealing with movements of people and vehicles so that many proven statistics, distributions, and parameters can be used in simulation.

Chapter 3

OVERVIEW OF MANET SIMULATION

3.1 Overview of Simulation

We have developed a realistic mobility and channel models. Figure 3.1 shows an overview of the steps for simulation. In Figure 3.1, five tools (i.e. Map builder, Process map data, Mobility simulator, Channel simulator1, and Channel simulator2) were implemented by our group. All tools are related to each other. The goal of those tools is to generate realistic mobility trace and corresponding pathloss trace data which are supported by QualNet simulator [32].

In this overview, I have contributed to parts of Process map data and Mobility simulator, and this thesis mainly focuses on those parts.

3.2 Map Builder

Based on real map or GIS datasets, a map builder draws realistic city graphs and generates map data that will be used in Process map data and Channel simulator1. For a building, there are seven building types, which are office, residence, store, office/store, residence/store, subway entrance, and parking lot. The map builder manually specifies one of the building types to each building. For an intersection, there are two intersection types, which are traffic light and normal. For a pathway, it can be roadway or a sidewalk. For each pathway the number of vehicle lanes and sidewalk lanes, if any, must be specified. Figure 3.2 gives some examples generated by Map builder.

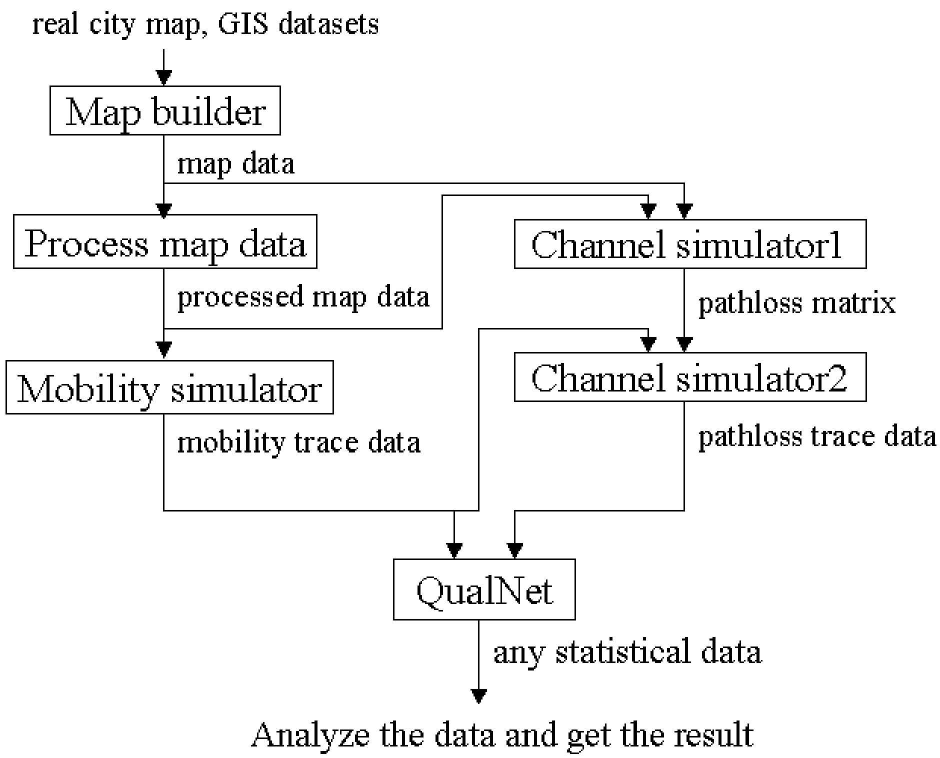

Figure 3.1: Overview of projects

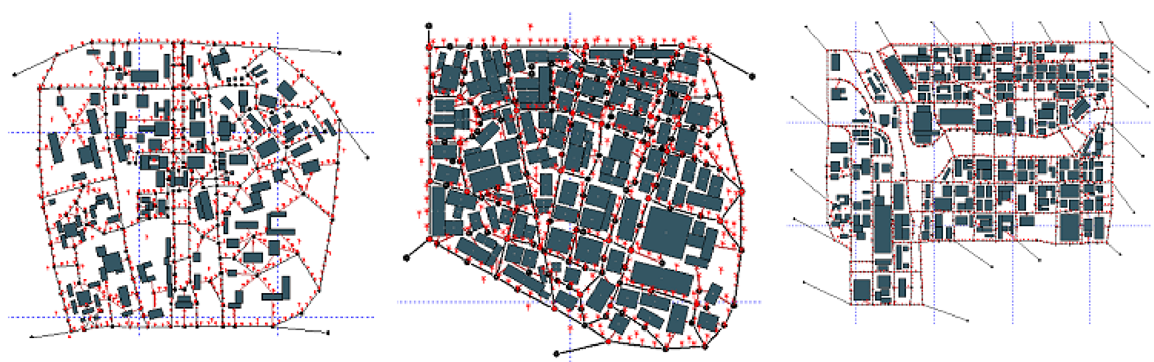

Figure 3.2: Example cities. Circles are intersections, lines are pathways, rectangles are buildings and red antenna shape is one of base station or fixed relay. Left : University of Delaware (size : 1421m by 1402m) Center : Paddington, London (size : 900m by 800m) Right : Chicago (size : 3200m by 2800m)

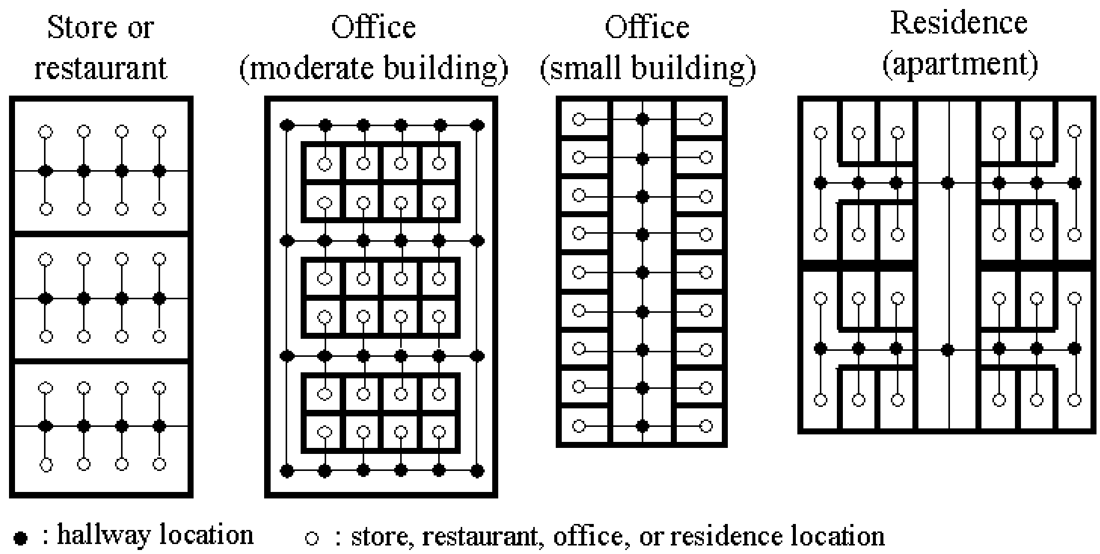

Figure 3.3: Locations in different types of buildings. Locations are marked with a circle while arcs are indicated by thin lines. The thick lines denote walls. The apartment building shown has four apartments each with five rooms. Size of the rooms is approximately constant. While larger buildings can accommodate more rooms

3.3 Process Map Data

This makes a city graph that consists of many vertices and arcs, and only along them can the MNs move (hence this is similar to a graph-based mobility model). A vertex can be the location of office, residence, restaurant, store, hallway, sidewalk, road, traffic light, and door. An arc can be hallway, walkway, sidewalk, crosswalk, and roadway. In the building, each location has route information to the door, and each door has route information to the other doors in the manner of shortest path. By using this route information, in a mobility simulator each MN can reach its destination at a certain location without computing the shortest path repeatedly.

According to building types, the interior layout is automatically generated. Figure 3.3 shows the interior layout [33]. If the building type is residence/store,

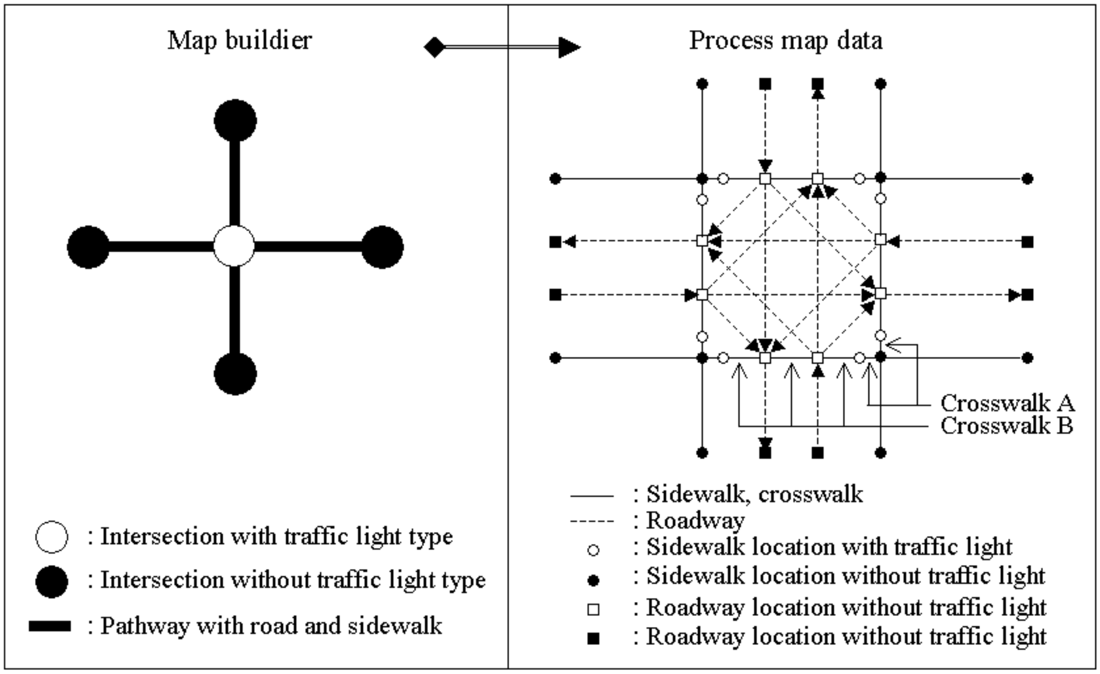

Figure 3.4: Process of pathway and intersection. Left : Map drawn by map building. Right : Process map data program figures out all vertices and arcs based on map data

stores or restaurants will be on the first floor and residences will be placed above the first floor.

One of the difficulties is to determine the locations of sidewalks, crosswalks and roadways from pathway and intersection information in the map data generated by Map builder. Figure 3.4 shows one of many cases in Map building.

In order to disallow nodes from passing through each other, a node can only pass another node if they are in different lanes. Sidewalks, crosswalks, roadways, and hallways have user defined number of lanes. However, at some locations, extra lanes are added. For example, at a crosswalk such as the one shown in Figure 3.4 has 16 lanes because many people crowd around the edge of the crosswalk (waiting for the green light) and in crosswalk.

Figure 3.5: Paddington city with normal view.

3.4 Mobility Simulator

By simply loading all graph and routing information from processed map data, the mobility simulator can only focus on mobility simulation, so it can achieve fast mobility simulation.

Figure 3.5, 3.6 show the Paddington city drawn by Open GL. MNs move along the graph shown in Figure 3.6 according to mobility models that will be discussed in Chapter 4. For each node, the complete movements are recorded during simulation. At the end of simulation, the mobility simulator generates mobility trace data for all nodes.

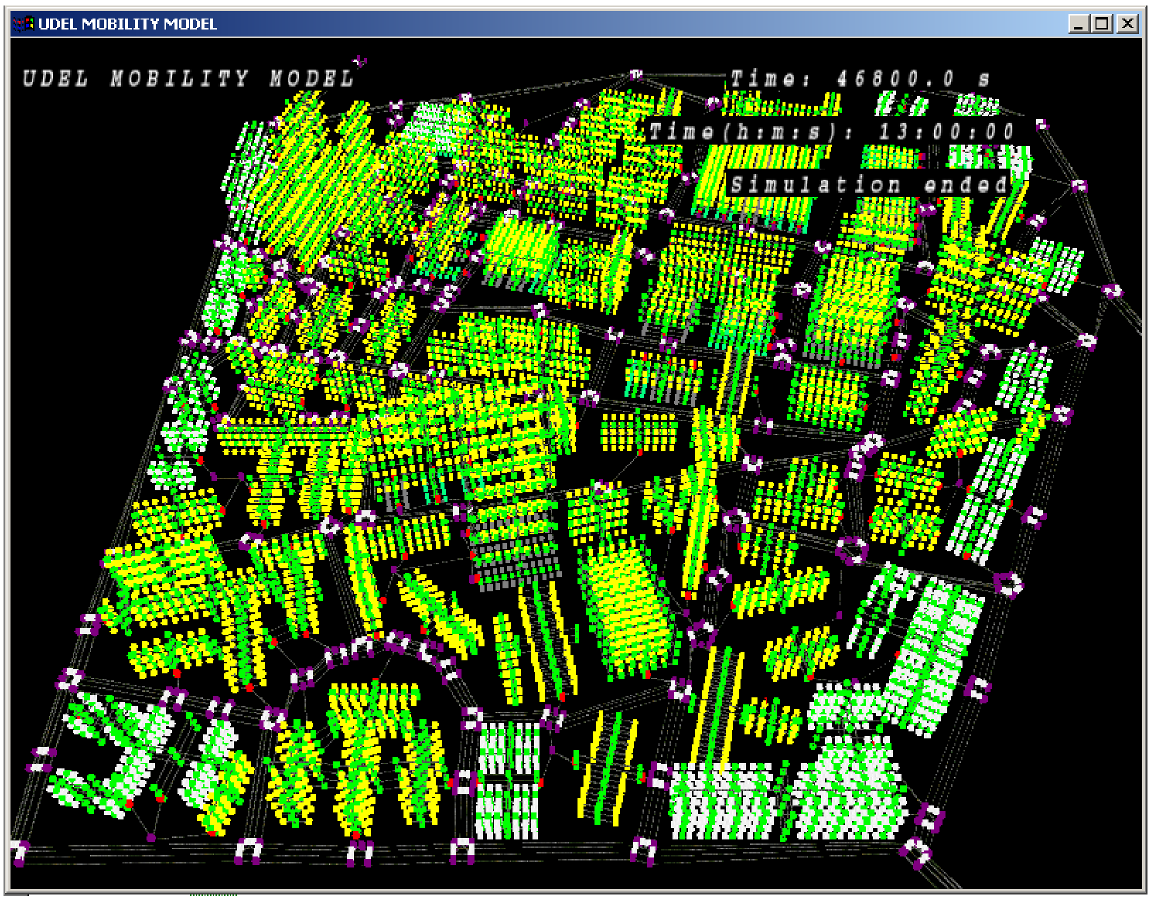

Figure 3.6: Paddington city with showing all vertexes and arcs. Yellow rectangle is an office location, white one is residence or roadway location, green is a hallway, purple is sidewalk, gray is a store, red is a door, and pale green is a restaurant.

3.5 Channel Simulator1

In the outdoors, reflection, transmission, and diffraction in 3 dimensions are modeled, while indoors, the attenuation factor models are considered [34]. The 2-D space is divided into a grid. Each square of the grid is called a floor-tile. To reduce computing power, only floor-tiles along the sidewalks, roadways, hallways, and crosswalks are considered. Through beam tracing technique in [34], pathloss of each floor-tile to all floor-tiles are computed. Since enormous computing power is needed to make the whole pathloss matrix, we developed a master-slave program. The objective of the master-slave program is to process a huge amount of work using many computers. The master computer generates many tasks and gives them to slave computers. Slaves get the tasks from master, and perform their work based on the received tasks, then send results back to the master or servers. While it takes several days to obtain pathloss matrix, this process is only needed once.

3.6 Channel Simulator2

For a particular mobility scenario traced by mobility simulator, this generates pathloss of each node to all other nodes per second by looking up the pathloss matrix, because we know the positions of all nodes per second from mobility trace data.

Figure 3.7 and 3.8 shows an example of pathloss [35].

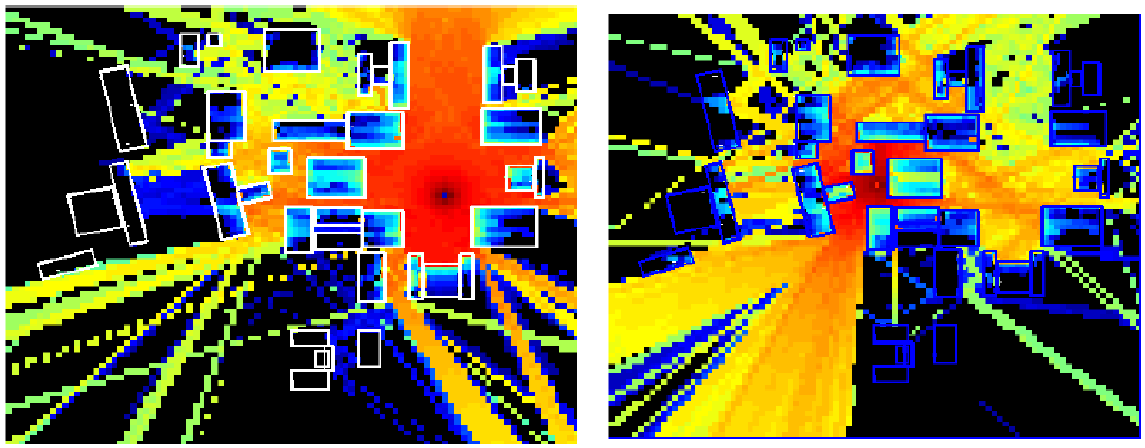

Figure 3.7: Path Loss from outdoors. The dark red indicates low path loss (high signal strength). The yellow indicates higher path loss and black indicates path loss over 100dB. The buildings are shown with a white outline on the left and a blue outline on the right [35].

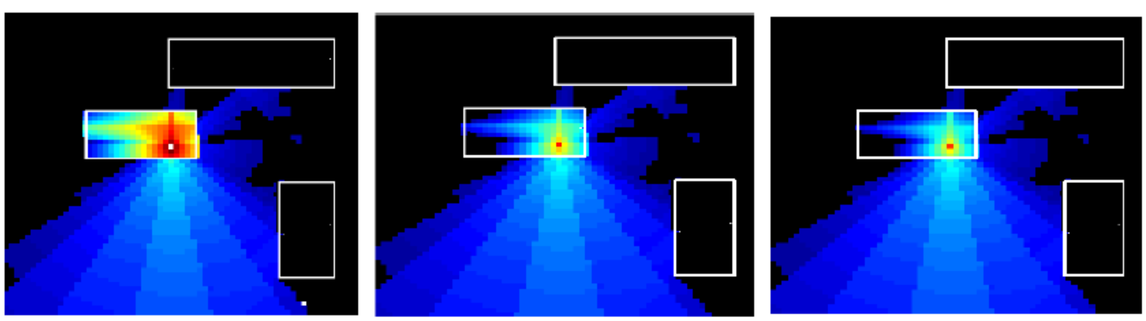

Figure 3.8: Path Loss from Inside a Building. The building is indicated with the white outline. The transmitter is on the first floor of the building. The figure on the left shows the loss on the first floor of the building as well as in the area around the building. The middle and far right plots shows the path loss on the second floors and third floors as well as in the area around the building [35].

Chapter 4

REALISTIC MOBILITY MODELING

4.1 Introduction

Mobility model is based on three mature research areas, urban planning [36], [37], meeting analysis [38], and use of time [39]. The resulting model is a three layer hierarchical model. The top layer is the activity model that determines the high-level types of activities as well as the time when people start and end the activities, and the location where the activity is performed. The data used to develop this model is from the recent US Bureau of Labor Statistics (BLS) use of time study [40]. This study includes interviews with roughly 20,000 people. Furthermore, the BLS determined weightings to account for over sampling of some types of people (e.g., unemployed people tend to be at home at the time of the interview call and tend to be over sampled). Hence, the significance of the study is more than that the 20,000 were actually interviewed. This study collected detailed data on the interviewee's day, and included the times that activities were started and stopped, where the activities were performed, and for what reason the activity was performed.

The second layer of the pedestrian mobility model is the task model. While performing a particular activity, a person may carry out many tasks. For example, the model discussed here focuses on office workers. While such nodes are performing a activity, there are two possible tasks, namely, working at their desk, and meeting with other workers. The basis of this part of the mobility model is several seminal studies of worker meetings performed within the management research community (see [38] and references therein). This part of the model makes it possible to determine how nodes move within a building and how nodes are clustered within buildings. Mobility within buildings is important if nets utilize relaying by mobile nodes. For example, an outdoor network such as Philadelphia's can greatly increase its indoor coverage if mobile nodes can act as relays [41]. To determine the performance of such relaying, the mobility of indoor nodes must be modeled.

The third layer of the mobility model is the agent model. Such agent models have been investigated within the architecture community, and define how nodes navigate walkways to their desired destinations. This model is based on urban planning research, especially the seminal work of Pushkarev and Zupan [36] as well as several other pedestrian mobility studies. A key feature of this part of the mobility model is that it realistically models how nodes form clusters or platoons. Such clusters are important since the proximity of nodes to other nodes plays an important role in the performance of mesh networks.

4.2 Activity Model

4.2.1 Pedestrian Activity Model

This part of the mobility model is based on the US Bureau of Labor Statistics 2003 time-use study [40]. This study identifies a large number of activities. We focus on those activities that indicate location and group activities together that are performed in the same location (e.g., all activities performed at home are grouped together into the at home activity). While the BLS study also collected coarse location information, both activity and location information were used to determine the location used in the modeling effort. We focus on eight types of activities: working, eating not at work, shopping, at home, receiving professional services, exercise, relaxing, and dropping off someone. Note that since we focus on location and mobility, eating at work is counted as work. Eating not at work includes eating at a restaurant and buying food somewhere besides at work. Shopping includes all types of shopping except buying food. Receiving professional services ranges from things such as getting medical attention to receiving household management and maintenance services not performed at home.

During the simulation initialization, each node is given an office and home. It is assumed that work is done within the building where the node's office is (work done at home is included in the at home activity). Eating is done at a restaurant (eating at home is included in at home activity). Shopping is done at one of many stores. Receiving professional service is done at an office that is not the node's office. Our current model does not specify a location for relaxing and dropping some one off. Dropping someone off includes meeting children at school and taking them home. For the purpose of mobility modeling, we model such activities as a trip home followed by a trip to an office location. The node remains at the office location until the drop off activity if complete. The relaxing activity is modeled as going to an office location (much like receiving professional service).

This model effort focuses on the work day which consists of being at home, going to work, working, and perhaps taking a break and returning to work and then leaving work and returning home. The model ignores activities before and after work. Future work will include the rest of the day.

For each person, the following steps are taken to determine the activities that they perform.

1. Select a home and office.

2. Determine the arrival time at work.

3. Determine the duration at work.

4. Determine if a break from work is taken. (The next 5 steps assume a break is taken.)

5. Determine the time the break starts.

6. Determine the number of activities performed during a break

7. Determine which activities are performed during the break.

8. Determine the duration of each activity.

9. Determine the arrival time back at work and determine if a break is taken again. If so, steps 5-9 are repeated.

Selection of home and office An office for each node is selected at random. Once an office is selected, a home is selected that is nearby the office. Specifically, a home is selected so that the distance from the home to the office matches the distribution shown in figure 4.6. This distribution is based on walking distances observed by Pushkarev and Zupan. The model also allows for nodes not to walk to work, but to arrive via the subway or car. Such nodes do not take breaks that involve going home. According to the BLS data, 90% of the people drive to work, 5% walk to work, the remaining 5% take other forms of transportation including subway, bus, and other forms of transportation to work. These parameters can be altered for the simulation of European cities that have significantly fewer people driving to work.

Arrival time at work Figure 4.1 shows the complementary cumulative distribution function (CCDF) of the time of arrival at work. The observed values were fitted with a mixture of exponential and Gaussian distribution. Specifically, with probability of 0.552, the time of arrival is normally distributed with mean 7:46 and standard deviation of 45 minutes. With probability $(1 - 0.552)$, the time of arrival is exponentially distributed with the mean time of arrival 12:00. The exponential distribution is shifted so that the minimum earliest time of arrival in this case is 5AM. The normal distribution is truncated so that no arrivals occur before 5AM.

Table 4.1: Duration at work model parameters

time	α	μ	σ	m
≤8AM	0.91	8:09	1:06	9:50
8-9	0.85	7:49	0:56	8:52
9-10	0.81	7:16	1:17	5:52
10-11	1.0	7:11	2:16	-
11-12	0.70	7:16	2:11	5:00
12-1	1.0	6:19	2:40	-
1-3	0.5	7:33	0:55	4:31
3-6	0.83	6:18	1:55	2:07
≥6	1.0	4:30	2:26	-

Duration at work Figure 4.2 shows the CCDF of the duration at work for people that arrive at work between 7 and 8 am and for those that arrive between 10 and 11 am. These distributions and those for other arrival times at work were fitted with a mixture of a normal random variable and an exponential random variable. These distributions have four parameters, α, the probability of selecting the normal distribution, μ and σ the mean and the standard deviation of the normal distribution and m, the mean of the exponential distribution. Table 4.1 shows the value of these parameters for the different arrival times at work. Surprisingly, while the model is simple, the fit shown in figure 4.2 is a typical quality of fit throughout the day. On the other hand, from Figure 4.1 it can be seen that the most important distribution is that for nodes arriving between 7 and 8 am.

Whether a break is taken The probability of whether a break is taken depends on the time of arrival at work. Note that if a break is not taken, the person may still eat lunch, but they do not leave the building. We fit the probability of

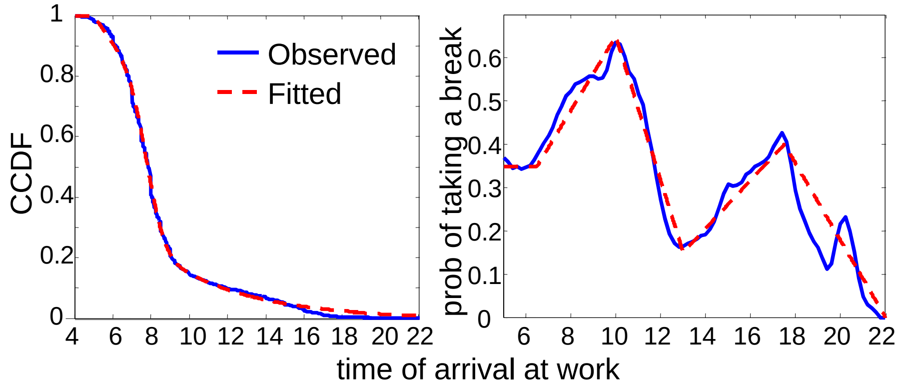

Figure 4.1: Left. Complimentary cumulative distribution function (CCDF) of the time of arrival at work. Right. The probability of taking a break given the arrival time at work. This includes arrivals after a break.

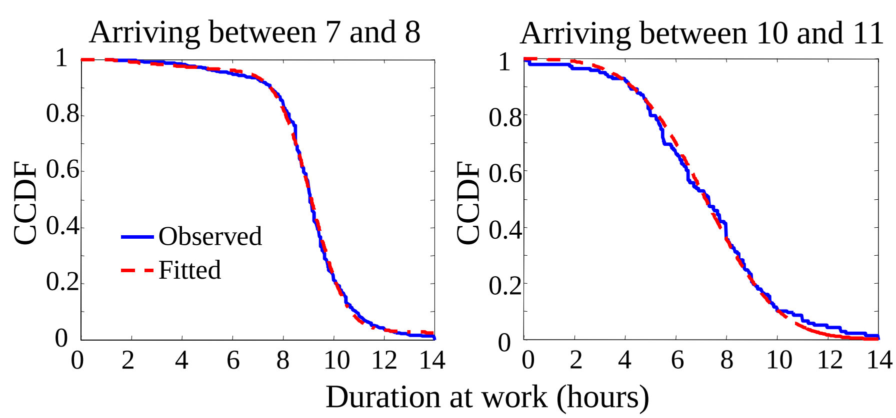

Figure 4.2: The CCDF of the duration at work for two different arrival times at work.

taking a break given the time of arrival with a piece-wise linear function.

$$P\left(\text{taking a break}|\text{ arrival time at work} = t\right)$$
$$= \begin{cases} 0.35 \text{ for } t < 6.5 \\ 0.86\left(t - 6.5\right) + 0.35 \text{ for } 6.5 \leq t \leq 10 \\ 0.17\left(t - 10\right) - 0.65 \text{ for } 10 \leq t \leq 13 \\ 0.056\left(t - 13\right) + 0.15 \text{ for } 13 \leq t \leq 17.5 \\ -0.08\left(t - 17.50\right) + 0.4 \text{ for } t \geq 17.5 \end{cases}$$

Note that this equation uses fraction of hours past midnight, not hours and minutes. This model and the observed probability is shown in Figure 4.1.

The time the break is started Clearly one cannot go on a break before they arrive at work. However, once they arrive at work, the rate that a person goes on a break does not depend on how long they have been at work. Figure 4.3 shows this rate conditioned on the person arriving at work one hour earlier, two hours earlier, and unconditionally. It can be seen that the duration at work has only a minor impact of the time to take a break and that this difference is within the confidence intervals. Thus, we assume that the rate of going on a break is independent of arrival time, assuming that the node has already arrived at work. The rate that a person takes a break is approximated by

$$r\left(t\right) =$$
$$\begin{cases} 0.004 \text{ for } t < 10.5 \\ 0.006 \times \exp\left(-1.7\left(12 - t\right)\right) \text{ for } 10.5 \leq 12 \\ 0.006 \times \exp\left(-0.6\left(t - 12\right)\right) \text{ for } 12 \leq t \leq 14 \\ 0.0058 \times \exp\left(-0.3\left(5 - t\right)\right) \text{ for } 14 \leq t \leq 18 \\ 0.0058 \text{ for } t > 18 \end{cases}.$$

By rate of taking a break, we mean that the probability that a node will take a break within the time interval from t_0 to t_1 is $\left(t_1 - t_0\right) \int_{t_0}^{t_1} r\left(\tau\right) d\tau$. The observed rate and fit is shown in Figure 4.3.

Number of activities performed during a break Figure 4.4 shows the probability of performing different numbers of activities during a break. We see that over the course of the day, the number of activities performed varies. However, the variation is small, and hence we take the probability to be independent of the time of day.

Which activities are performed during a break The types of activities performed during a break strongly depend on the number of activities to be performed. Figure 4.4 shows the fraction of breaks that include the indicated activity. Note that if more than one activity is performed, the fractions sum to more than one.

The duration of activities The time spent performing an activity depends on the type of activity. Figure 4.5 shows the CCDF of the duration of three activities. The distribution of the duration of eating shows a jump at 1 hour. Smaller jumps are noticeable in the distribution of other activities. The duration of these and the other activities are modeled as a mixture of an exponentially distributed random variable conditioned on the duration being larger than a minimum duration along with deterministic duration of one hour. Thus, the distribution of the duration of each activity has three parameters, μ, the mean of the exponential distribution, $\underline{d}$, the minimum duration, and ρ, the probability of the duration lasting exactly one hour. Table 4.2 shows the value of the model parameters for the different activities considered.

Once the activity has been selected, the location of the activity must be determined. Specifically, eating requires selecting a restaurant, exercising requires selecting a gym, getting professional service requires selecting an office location, shopping requires selecting a store, dropping someone off requires selecting an office location to drop them off at.

We assume that people walk to the location that is required to perform

Table 4.2: Duration of activity model parameters

activity	μ	$\underline{d}$	ρ
eat	0:31	0:20	0.18
shop	0:28	0:20	0.03
at home	1:00	0:20	0.12
professional	0:44	0:10	0.04
exercise	0:35	0:20	0
relax	0:27	0:15	0.01
drop-off	0:19	0:10	0.02

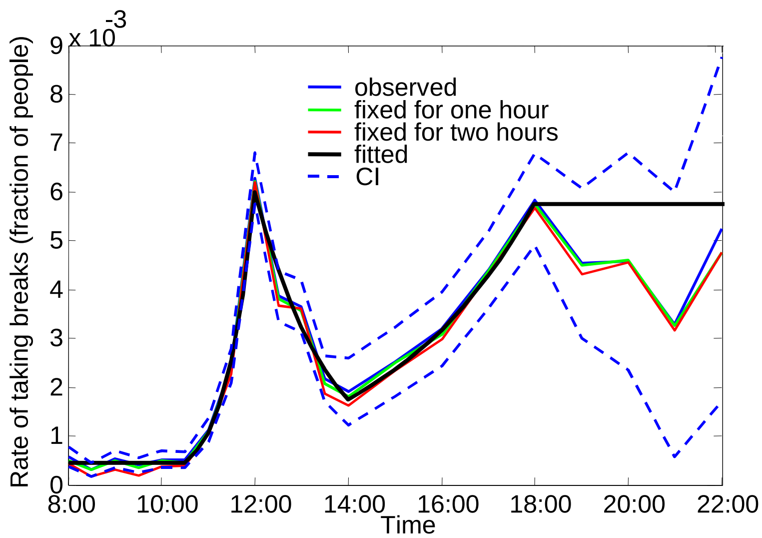

Figure 4.3: The rate that a person takes a break and leaves work given the current time. Also shown are the rates conditioned on the person being at work for at least one and two hours. These rates are within the confidence intervals that are also shown. Finally, the fitted rate is also shown.

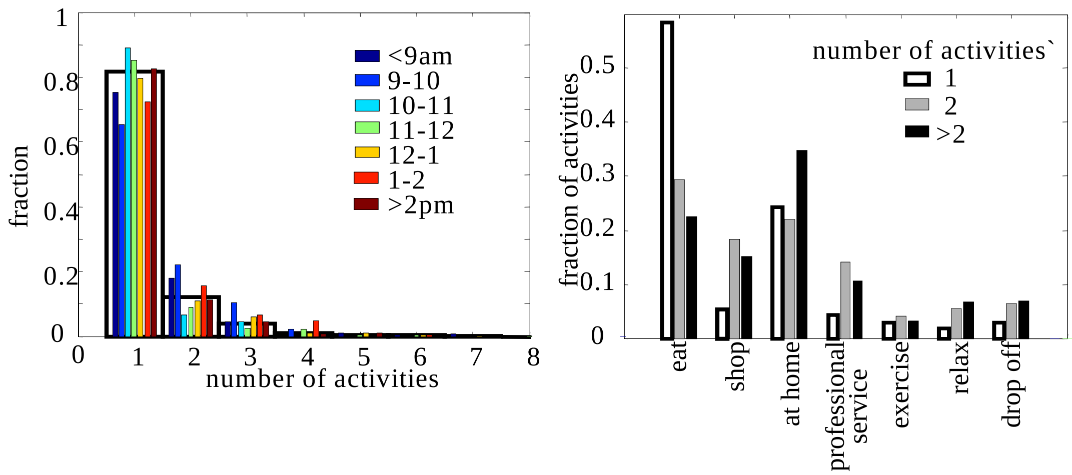

Figure 4.4: Left: the number of activities done during a break conditioned on the time that the break is started. Right: the fraction of time that a break includes the indicated activity given the number of activities performed within the break.

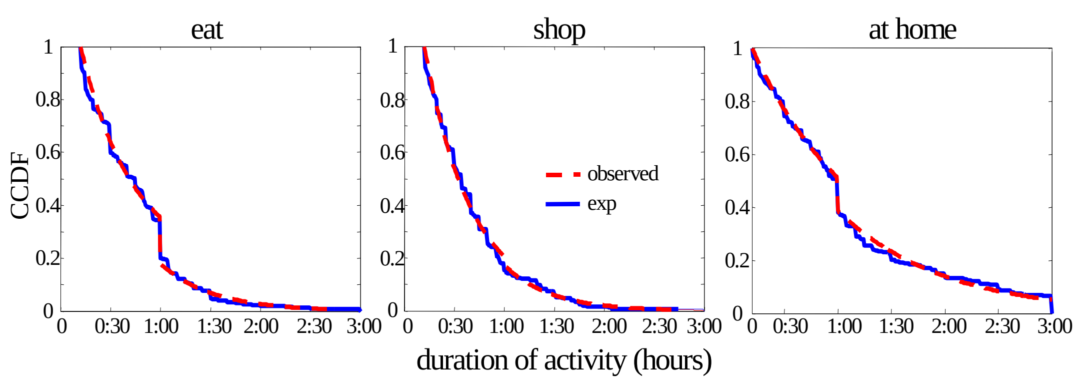

Figure 4.5: CCDF of the duration of eat, shop, and at home activities.

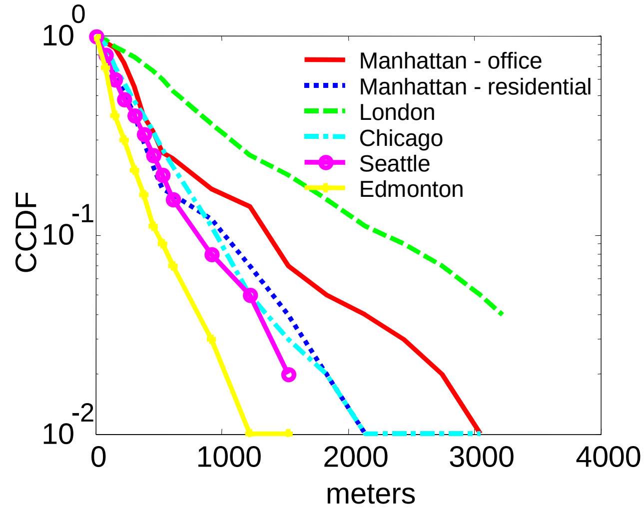

Figure 4.6: CCDF of Distance Traveled During Outdoor Walking Trips. This data is from [36].

the activity. Future work will include the case where people take other forms of transportation. Pushkarev and Zupan [36] observed the distribution of the distance that pedestrians walk (see Figure 4.6). We see that the distance is well modeled by an exponential distribution with means 554 m, 380 m, 403 m, 344 m, 813 m, and 216 m for Manhattan from office buildings, Manhattan from residences, Chicago, Seattle, London and Edmonton respectively. We see that the US cities have approximately the same mean. Thus, we select a location of the correct type (e.g., a store for shopping) at random such that the walking distance is exponentially distributed with mean 400 m.

Groups of pedestrians play an important role in platooning [36]. Again, there is little data on the frequency of groups. However, we have made observations of over 500 pedestrians in an urban street and found the number of pedestrians within a group is well modeled with the Zipf distribution with shape parameter of 2.18, i.e., $P\left(\text{Group size} \geq g\right) = 1/g^{2.18}$. We allow groups of nodes to congregate in an office and then proceed to a destination. While in transit the nodes walk abreast of each other unless there are on-coming nodes or the sidewalk cannot support all nodes in the group. In such cases, some nodes will follow behind. The speeds to the group members is forced to be the same and the nodes cross intersections with the grouping intact. The dynamics of groups acts to block other nodes from passing as observed by [36]

4.2.2 Vehicle Activity Model

Traffic simulators such as CORSIM [42] allow vehicle trips to be generated in two ways, with orgin-destination (O-D) flow matrices or with turning probabilities. With O-D matrices, the rate at which vehicles enter the simulated region at a origin O and proceed to the destination D is given by the O, D element of the O-D matrix. If only turning probabilities are used, vehicles enter the modeled area at one of a preselected locations and proceed until the vehicle arrives at any exit location (often at the edge of the modeled area). At each intersection, vehicles turn or go straight according to the turning probabilities assigned to that intersection. O-D matrices yield a more accurate simulation, however, accurate O-D matrix are difficult to determine, whereas turning probabilities can be determined by simply counting vehicles turning at each intersection. Thus, both approaches are used for urban traffic engineering.

Drawbacks of turning probabilities are that vehicles might travel in long loops or meander through the city for extended periods of time. However, since turning probabilities are quite small (often they are in the range of 0.3 to 0.1 [43]) such unrealistic behavior is rare; most trips proceed through the city with only a few turns. Our simulator currently uses homogeneous turning probabilities. Exit and entry points are defined by the map as described in Section 3.2.

To model the rate that vehicles enter the city, we borrow urban traffic models for "upstream" lights (i.e., the traffic that exits a light upstream of the light under investigation). The upstream traffic is from two sources, vehicles that pass through the green light and go straight, and vehicles that turn on to the street. Following [44], it is sufficient to assume that the number of vehicles that enter is Poisson with mean

$$\frac{\lambda_{VehicleStartRate} \times \text{Signal Period} \times (1 - prob\_turning)}{\text{Number of Entering Roads}} \tag{4.1}$$

assuming that the number does not exceed the number that can pass through an intersection during a single green light. These vehicles enter at periodic moments with periods equal to the traffic signal period. Furthermore, the number of turning vehicles into the road that leads to the modeled area is Poisson process, but with rate

$$\frac{\lambda_{VehicleStartRate} \times prob\_turning}{\text{Number of Entering Roads}}. \tag{4.2}$$

Hence, the total average rate that vehicles enter the city is $\lambda_{VehicleStartRate}$. The realistic values of $\lambda_{VehicleStartRate}$ are not found in the literature.

4.3 Task Model

Some activities consist of a single task. For example, eating consists of going to a restaurant. However, shopping and working consist of multiple tasks. We model shopping as a simple random walk inside the store. However, work is modeled in a more complicated manner that focuses on modeling meetings. Specifically, [38], [45], [46] have collected data on the frequency, size, and durations of meetings; [45] includes two person meetings. These studies allow the model to include worker interactions. Thus, we model mobility while at work as a sequence of meetings followed by working in the node's office. This process repeats until the work activity is complete.

More specifically, meetings are simulated as follows. The time between meetings is assumed to be exponentially distributed. When a meeting begins, a random number of people is selected to attend the meeting. Based on the number of people

Table 4.3: Meetings model parameters

meeting size	mean duration	prob.
2	21 (min)	0.65
3	19	0.12
4	57	0.04
5	114	0.02
6	37	0.04
7	50	0.03
8	150	0.01
9	75	0.02
10	150	0.01
15	30	0.025
20	30	0.025

attending, the duration of the meeting is determined. The duration is assumed to be exponentially distributed.

The model parameters of the model are the mean time between meetings, the distribution of the size of meetings, and the relationship between number of meeting participants and the mean meeting duration. These parameters are determined from [38], [45], [46] . Specifically, the mean time between meetings is 18 minutes while Table 4.3 gives the remaining of the model parameters.

4.4 Agent Model - Node Dynamics and Interactions

Since the pioneering work of Pushkarev and Zupan [36], it has been known that pedestrians are not uniformly distributed but tend to be grouped into clusters or, in the terminology of urban planning, platoons. Since the distribution of nodes plays an important role in the performance of mesh networks, the mobility must also model platoons. This part of the model is known as the agent model and is responsible for determining the trajectory of the node as it moves from one location to the next. In our simulator, nodes take the shortest path, hence path finding is not an important part of the agent model. Rather, the agent model focuses on the dynamics and interaction between moving people. More specifically, the agent model consists of enforcing a distance-speed relationship between nodes and lane changing rules. These are discussed in the next two sections. In Chapter 7, the model is validated by comparing the size of platoons created by the model to those observed by Pushkarev and Zupan.

4.4.1 Inter-node Speed-Distance Relationship

When a node with a higher desired speed catches up to a slower moving node, it will either follow or pass. To understand the dynamics of catching up, it is necessary to understand the distance-speed relationship. The impact of this relationship is that nodes will be tightly packed (i.e. high density) if their speed is low (congestion), but if the speed is higher, then the nodes must be further apart (low density). Since the density of nodes plays an important role in the performance of mesh networks, the distance-speed relationship must be understood and realistically modeled. For vehicles, the distance-speed relationship, which we denote as $D(S)$, is closely related to the "two-second rule" that specifies that a following vehicle should not be closer than two seconds behind the vehicle it follows. For both vehicles and pedestrians, these relationships have been extensively studied.

The distance-speed relationship for pedestrian is studied in [47] and [48]. Figure 4.7 shows the distance-speed relationship derived from these observations[1]. We approximate this relationship with

$$D(S) = S^* D_{\min} / (1.08 \times S^* - S) \tag{4.3}$$

where $D_{\min}$ is the minimum distance between people without touching and S^* is the desired speed of the pedestrian. $D_{\min}$ was found to be at least 0.35m [36].

In the case of vehicles, the distance-speed relationship depends on weather conditions (e.g., dry vs wet road), on the traffic regime, and the recent traffic regime

[1] The plot shown is based on area-speed relationships with the assumption of 0.75 meter of lateral space between people as found by Oeding [49].

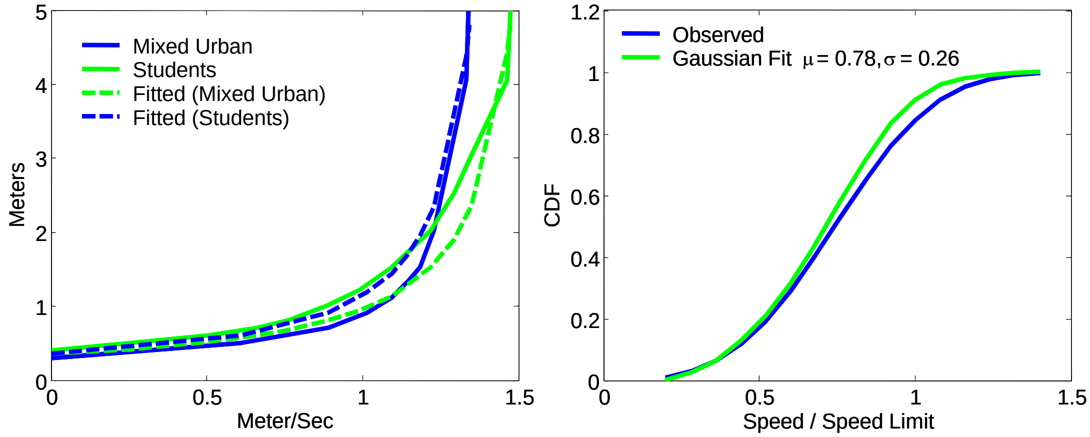

Figure 4.7: Left: Distance-Speed Relationship for Pedestrians. The mixed urban pedestrian data is adapted from [47] and the student observations are adapted from [48]. Right: CDF of the ratio of observed speeds to speed limit and the CDF of a fitted Gaussian distribution.

history [50], [51]. While factors such as recent traffic regime history are important in highway traffic, we focus only on the more simple urban street traffic and use the speed-distance relationships observed at low speeds which such factors are less noticeable [51]. While $D(S)$ is not exactly linear, it is often modeled as linear, specifically,

$$D(S) = \alpha + \beta S \tag{4.4}$$

In [50], (α, β) were found to be $(1.78, 10.0)$ and $(1.45, 7.8)$ in dry conditions and $(0.415, 8.3)$ and $(0.230, 6.0)$ in wet conditions. Here, and throughout the next sections distances are in meters and speeds are in meters/sec. These values of α and β are in agreement with the observations presented in [51] and [52].

It has been found that speeds desired by pedestrian are approximately Gaussian with mean 1.34 m/s and standard deviation 0.26 [53], [54], [55]. For vehicles, the ratio of the vehicle's speed to the speed limit presented in [56] can be modeled as Gaussian with mean 0.78 and standard deviation 0.26 (see Figure 4.7).

4.4.2 Lane changing

While traffic lights are an important cause of platooning, lane changing also plays an important role [36]. A nodes will certainly not pass if there is no room (e.g., if the other lanes are full). Even if there is room, both pedestrian and vehicle nodes might not pass out of choice, or select to slow down and follow the node ahead [57]. Such decisions lead to platooning.

Lane changes are grouped into two categories, discretionary and mandatory. The later category results when the node's current lane ends or is blocked by a fixed obstruction, or the route to the destination requires changing lanes (e.g., to exit or make a turn). For MANET simulation, the dynamics of mandatory lane changes can be ignored since the exact moment when the node does change lanes will not have a significant impact of the distribution of nodes and will only have a minor impact on position (and hence a minor impact on signal propagation to and form the node).

Since discretionary lane changing is related to platooning, it must be included in MANET mobility simulation. Discretionary lane changing depends on the difference between the speed that results from not changing lanes and the speed that could be achieved if a lane was changed as well as on other factors such as the presence of large vehicles [58]. We focus only on the speed aspects of discretionary lane changing.

In [58], the probability of changing lanes was modeled as

$$P\left(\text{desire to change lanes}\right) = 1/\left(1+\exp\left(A+B\left(V_{*}-V^{*}\right)\right)\right) \tag{4.5}$$

where V_{*} is the speed that the node would achieve if it remains in the current lane and V^{*} is the speed that would be achieved if the node changes lanes. Since speeds may show short-term variation, instantaneous determinations of V_{*} and V^{*} lead to erratic behavior. Instead, letting ν denote the node that is considering changing lanes, we define V_{*} to be the average speed of all nodes between ν and the next intersection, and V^* to be the minimum of the desired speed of ν and the average speed of the nodes in the target lane that would be between ν and the next intersection.

According to the findings of [58], if a node catches up to another node and there is room, it will change lanes 50% of the time when the speed difference is $V_* - V^*$ is zero. Furthermore, when the speed difference reaches one standard deviation of the node's speed distribution and there is room, the node will change lanes 66% of the time. To mimic this behavior at the slower speeds of urban vehicles and pedestrians, we set $A_{Vehicle} = -0.225$, $B_{Vehicle} = 0.1$, $A_{Pedestrian} = -0.225$, and $B_{Pedestrian} = 1.7$. While the Highway Capacity Manual suggests that pedestrian mobility is similar to vehicle mobility, it is not clear that the same model for making passing decisions can be used (albeit with scaled parameters). We examine the impact of these parameters in Chapter 7 and find that they do result in platooning that has been observed.

Other dynamics of lane changing include the selection of an acceptable gap between cars to change lanes into. It is not clear what the benefit of precise dynamics of gap acceptance would be for MANET simulation. Hence, we simply allow nodes to change lanes if changing lanes would not greatly disrupt the nodes in the target lane.

Since sidewalks are bidirectional, pedestrians may change into left-hand lanes and block on-coming pedestrians. Thus, we force all pedestrians that are in left-hand lanes to immediately change to a right-hand lane when confronted with an on-coming pedestrian. Furthermore, a pedestrian does not change into a left-hand lane unless there are no on-coming pedestrians from the pedestrian's position to the next intersection.

Chapter 5

MOBILITY SIMULATOR DESIGN

5.1 Introduction

Simulation is the technique of imitating the behavior of a real system [5]. In mobility simulation, the real system is the movements of real people and vehicles. By adequately understanding and thoroughly analyzing the phenomenon of the real system, we derive statistics, parameters, and models related to the real system as done in Chapter 4. By applying those at an appropriate moment during simulation, we can obtain the similar behavior of a real system.

Discrete event simulation (DES) is one way of allowing us to apply derived statistics, parameters, and models to simulation. At first, we investigate necessary events from the real system so that models can be applied at the proper moment. Whenever these events occur in the simulation, models are applied to simulation and a new event that will occur in the near future is computed and is inserted into event list in order by event occurring time. The discrete sequence of simulation time is determined by the event occurring time of each element of event list.

There are two types of DES : one is simple DES and the other is DES with reschedule. The routine of simple DES is as follows.

1. Make initial event for each mobile node.

2. Insert initial event to event list.

3. Get next event from event list.

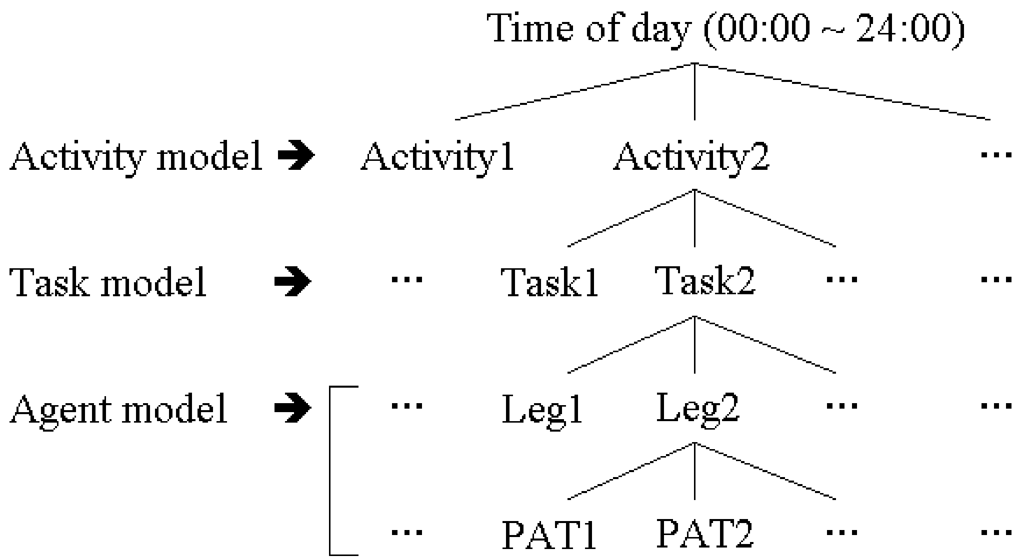

Figure 5.1: Hierarchical design

4. Process the event and generate new event(s).

5. Insert new event(s).

6. Repeat step 3.

The routine of DES with reschedule is the same as simple DES except for step 5 which should be replaced as Insert new events(s) or reschedule the event(s) already in event list (i.e. eliminate the existing event(s) and insert new event(s)).

Mobility simulator is based on DES with reschedule.

5.2 Implementation Methodology

Figure 5.1 shows how models derived in section 3 are applied to simulation. As described in the previous chapter, according to time of day, each node has its own list of activities (e.g. home, work, eat, work, and home). Furthermore, each activity has a series of tasks. For example, work activity will be made up of a series of "working at home office" and "meeting" task. The implementation makes each task into *legs*. We use the term leg since these are legs of the node's journey. A leg always have a start and end vertices as well as start time and end time. In the case that the leg has the same starting and ending vertex, we call the leg a waiting leg. Otherwise, the leg is a mobile leg.

Along a leg, there is a sequence of points and times (PAT). Each PAT entry is similar to a leg, it has starting and ending points and starting and ending times. However, the starting and ending points are within the leg. Each PAT entry holds the information of when a node is moving with a constant velocity. Thus, if a node begins a leg, but at the midpoint of the leg runs into heavy traffic and it forced to stop, and then, after the traffic has subsided, begins moving again to the leg end point, then the leg would have three PAT entries (two mobile PAT entries and one waiting entry). A generalization of the PAT would allow the node's acceleration to be constant during a PAT entry. However, this was not yet implemented.

5.3 Discrete Events

While each PAT represents a period of constant velocity, the reason for the change in velocity may vary. These reasons for changing velocity make up the events that must be simulated. There are six types of events used:

- REACH_EOS – a node reaches the end of arc.
- CATCH_UP – a faster moving node catches up to a slower moving or stationary node.
- EXIT_FIFO – a node is ready to enter a new arc.
- START_UP – a stationary node begins to move after catching up to a stationary node.

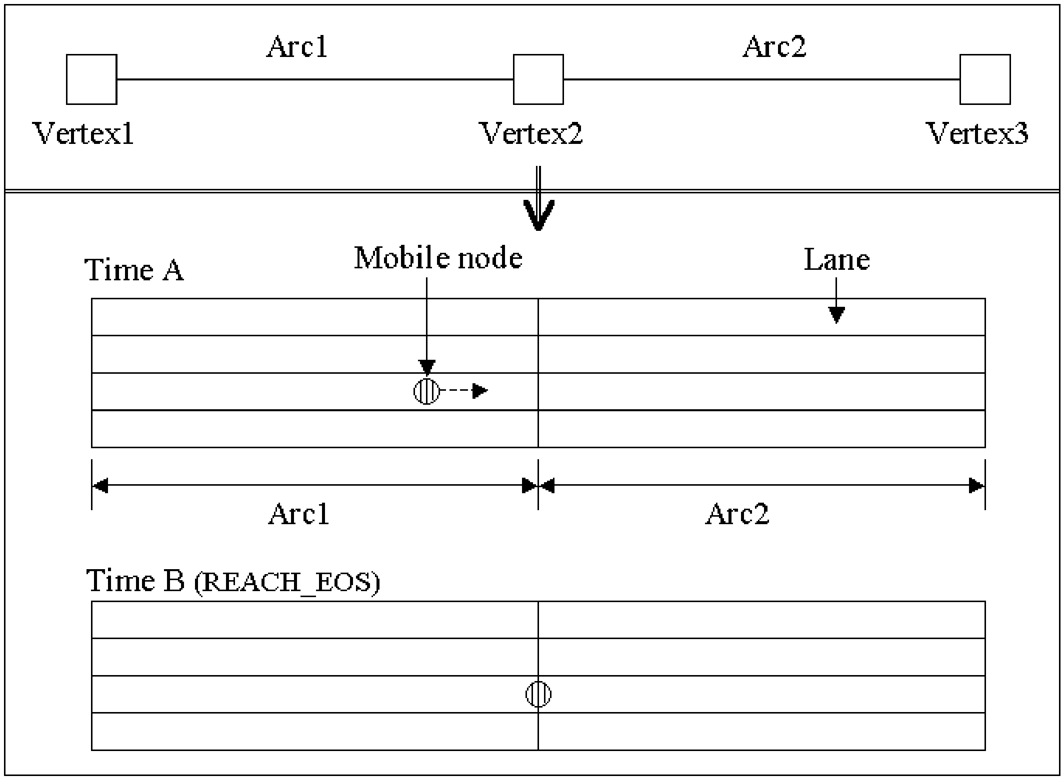

Figure 5.2: REACH_END_OF_SEGMENT event.

- MEET_IN_OPPOSITION – two nodes that are sharing the same lane but moving in opposite directions meet.
- SEND_NEXT_CARS – a car enters a road that leads to the simulated area.

Before describing these events, we discuss the concept of arc. General vertices and arcs can be modeled to the lower part shown in Figure 5.2. In Figure 5.2, an arc can be one of sidewalk, roadway, hallway, or crossway and mobile node can be one of person or vehicle. However, when a node is at some location along an arc, it is assigned to a particular lane. Depending on the user configuration, an arc can have two or more lanes. Note that an arc must have at least two lanes to support two-way traffic. In Figure 5.2 there are four lanes on Arc1, the arc between Vertex1 and

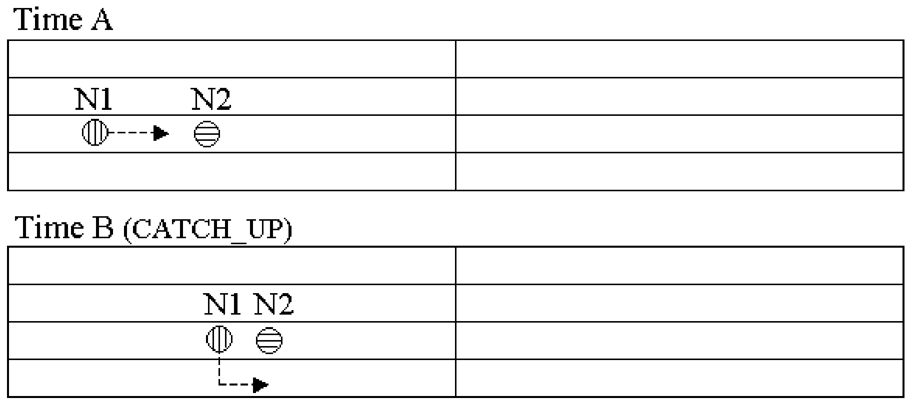

Figure 5.3: CATCH_UP event.

Vertex 2. When the MN reaches right end of arc1 (i.e. at time B), REACH_EOS event will occur.

In Figure 5.3, N1 is faster than N2, and N1 catches up to N2 at time B. It is important to note that when N1 catches up to N2 it is not at the same physical location, but the distance between N1 and N2 satisfies the distance-speed relationship in Eq. (4.3) or Eq. (4.4), where the speed is the speed of N2. In processing this event, N1 decides whether or not it changes its current lane to overtake front node N2. If there is enough space in the changing lane, N1 has a chance to change lane with probability of Eq. (4.5). If N1 does not change lanes, then N1's speed is set to the speed of N2. Thus, N1 and N2 will satisfy the distance-speed relationship. If N1 changes lanes to pass N2, then it moves at its desired speed.

At an intersection, a node may have to wait before entering the next arc. This waiting is done with a logical FIFO buffer. In Figure 5.4, we assume that each segment has one lane and that the speed of all nodes is the same. Each lane has two FIFOs: one for each end of the lane. The logical FIFO shown in Figure 5.4 is the left end of lane on Arc. At time A, a REACH_EOS event occurs as N4 reaches the left end of Arc. At time B a REACH_EOS event occurs as N5 arrives

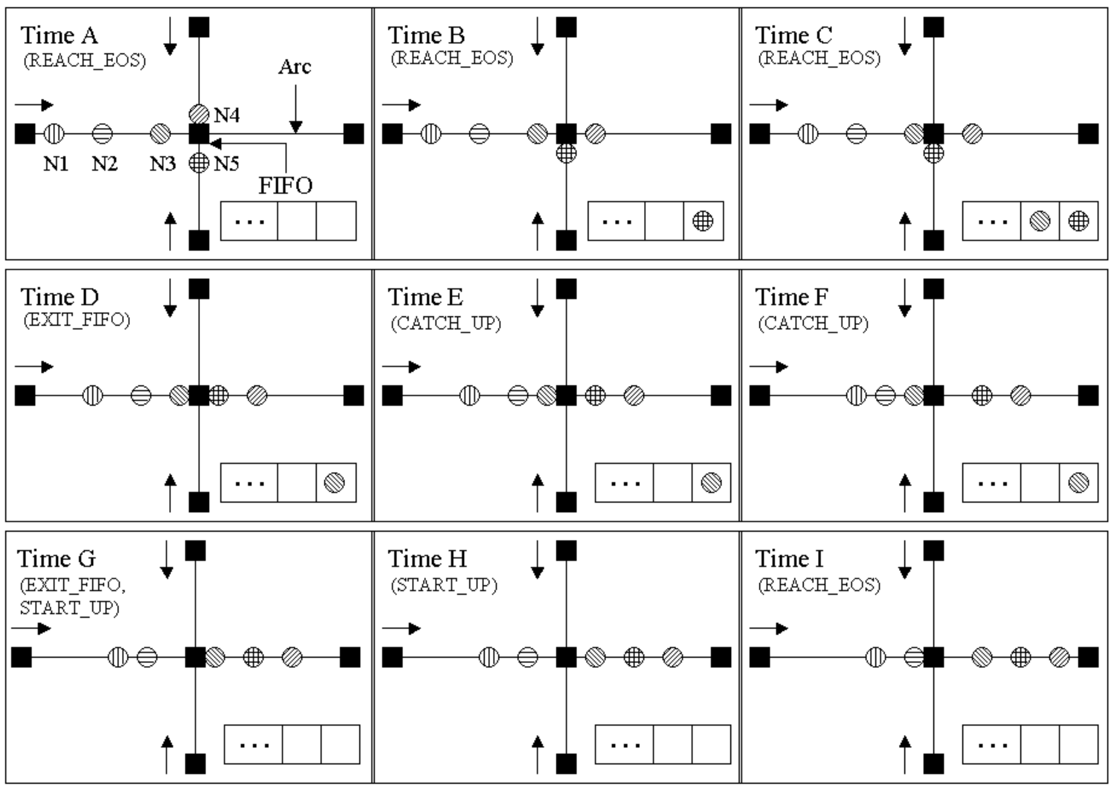

Figure 5.4: EXIT_FIFO, START_UP events.

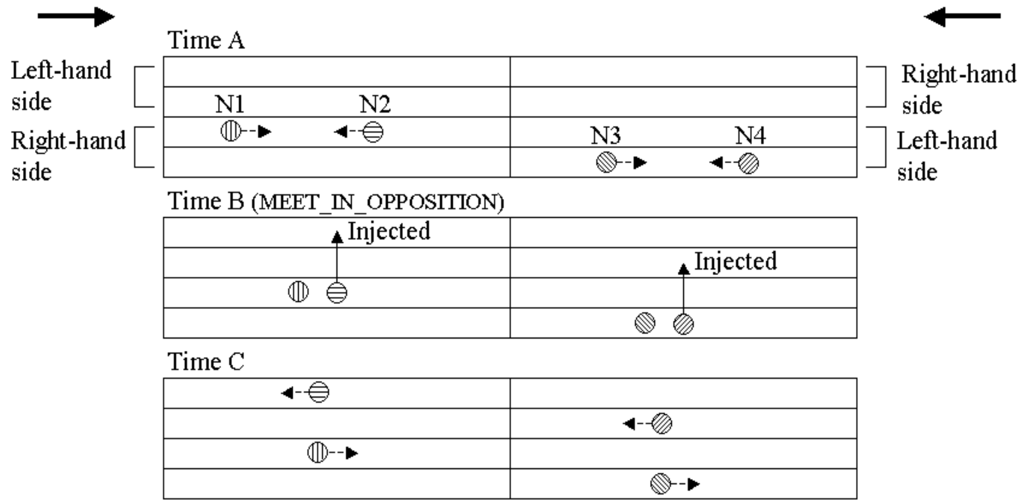

Figure 5.5: MEET_IN_OPPOSITION event.

at the left end of Arc. Since there is not enough space to enter Arc, N5 goes into the logical FIFO while physically begin stopped at the end of its current arc. At time C a REACH_EOS occurs as N3 enters the FIFO after N5 due to the same reason, and is physically stopped at the end of its current arc. At time D, since there is enough space, which is based on distance-speed relationship, for node N4 to enter Arc, an EXIT_FIFO event occurs and N4 exits the FIFO. At time E, a CATCH_UP event occurs as N2 catches up to N3, which is still stopped. Thus, node N2 is not placed into the logical FIFO but N2 is a short distance behind N3. This distance is given by the distance-speed relationship for a zero speed. Similarly, at time F, a CATCH_UP event occurs as N1 catches up to N2 and stops. At time G, an EXIT_FIFO event occurs and N3 exits FIFO and enters Arc. When exiting FIFO, if a node stopped behind, the node (i.e. N2) should start to move. Thus the EXIT_FIFO event causes a START_UP event. At time H, another START_UP event occurs and N1 starts to move. At time I, a REACH_EOS event occurs as N2 reaches Arc.

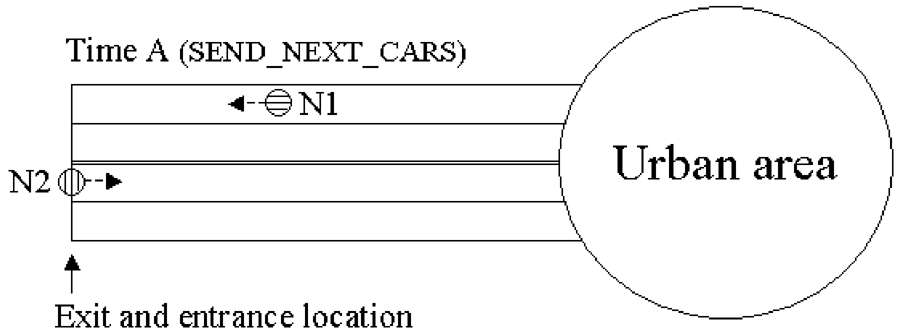

Figure 5.6: SEND_NEXT_CARS event.

In Figure 5.5, a MEET_IN_OPPOSITION event occurs at time B. Note that each lane is bidirectional. From N1 and N3's point of view, the right-hand side is lower two lanes, whereas from N2 and N3's point of view, the right-hand side is upper two lanes. At time B, N1 and N2 meet in opposition as well as N3 and N4. The rule is that when two nodes meet in opposition, the left-hand side node gives a way to right-hand-side node, so the left-hand side node is *injected* to the right-hand-side lane that is the same number of lane on the left-hand side (see Figure 5.5). Thus, N2 and N4 give way to N1 and N3. In the case of vehicles, this event never happens. If there are nodes in the lanes that N2 is injected into, then there will likely be some CATCH_UP event to stop the node that is behind the point where the N2 was injected. If the lane is very congested, then this CATCH_UP event may cause a ripple of CATCH_UP events. Nonetheless, the nodes in the lane make room for N2.

During simulation, cars will exit or enter the urban area. When it is time for some cars to enter the urban area, SEND_NEXT_CARS event occurs. The number of cars that enter the urban area per traffic signal period is Poisson distributed with mean given by Eq. (4.1). In Figure 5.6, N2 enters urban area at time A. It means N2 appears in modeled area. Some time later, N1 will exit the urban area. It means

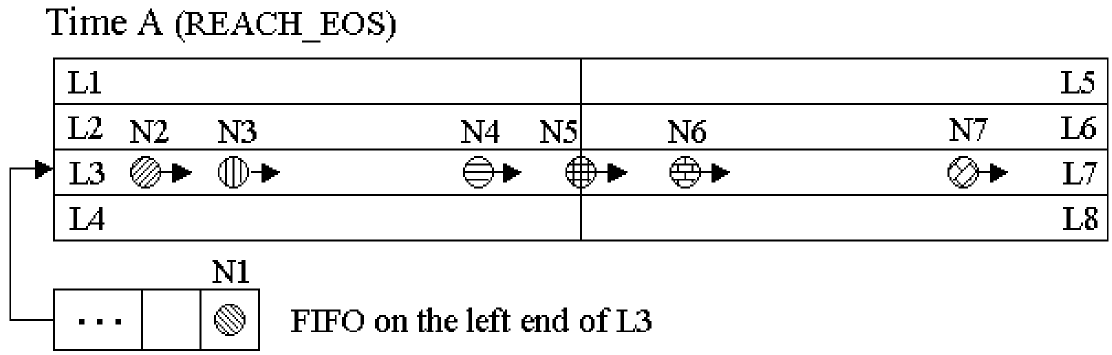

Figure 5.7: An example of REACH_EOS event

that N1 will disappear from the modeled area.

When REACH_EOS occurs, what should we consider as the next event? In Figure 5.7, at time A, N5 reaches the right end of L3. The next event on lane L3 is determined by examining which of the possible next events will occur first, where, in the case, there are four possible next events, namely N4 experiences a REACH_EOS event, nodes N3 experience a CATCH_UP event, N2 experiences a CATCH_UP event, N1 experiences an EXIT_FIFO event. The next event on L3 is inserted into the event list such that the list of events is ordered by the time of the event. After determining the next event on L3, N5 chooses one lane to enter and puts itself to the chosen lane. In this example, N5 chooses L7. Finally, the next event on lane L7 should be also redetermined by examining the possible events. The next event on Lane L7 may be that N7 will reach the right end of L7. However, once N5 has entered the lane, the next event on L7 may be that N5 will catch up N6. If that is the case, the previous event on L7 should be erased from event list. The newly found next event on L7 is inserted into event list. The other cases of events can be referenced in code on the website (http://www.eecis.udel.edu/~bohacek/UDelModels).

Chapter 6

SIMULATIONS

Four simulations are performed. Each section explains simulation environment. To see the results of each section, you can directly jump to Chapter 7.

6.1 Traffic Light

This simulation tries to show the behavior of pedestrians at the traffic light, so a crosswalk is zoomed in. The right side of Figure 3.4 explains how the crosswalk is constructed. Note that the number of lanes in the crosswalk is 16. Table 6.1 shows parameters used for simulation.

6.2 Platooning

This simulation tries to show that people make platoons due to group mobility and inter-node dynamics (i.e. lane changing decision). Both constrained random waypoint model, which is the same as random waypoint, but nodes only move through the graph and there are no inter-node dynamics, and UDel mobility model are simulated. Table 6.2 is used as parameters.

Table 6.1: Parameters for traffic light simulation

Number of pedestrian nodes	5,000
Simulation time	1,500 seconds
Map	University of Delaware
Traffic light period	90 seconds

Table 6.2: Parameters for platooning simulation

Number of pedestrian nodes	3,000
Simulation time	2,000 seconds
Map	University of Delaware
Traffic light period	90 seconds
Mobility model	constrained random waypoint, UDel mobility model

Table 6.3: Parameters for Paddington city simulation

Number of pedestrian nodes	10,000
Number of cars	100
Number of UAV	2
Simulation time	7,200s (11:00 ~13:00)
Map	Paddington city
Traffic light period	90s

6.3 Paddington City

This simulation is to show movement of pedestrians, cars, and UAVs (unmanned aerial vehicle) in the city. Actually, UAV mobility modeling is underway. Simulation mainly considers lunchtime.

6.4 Validation of Pedestrian Mobility

This simulation is for validation of pedestrian mobility model and is performed with a variety of configurations shown in Table 6.4 including counting pedestrians on a block with and without buildings, various sizes of sidewalks (from 4 lanes to 32 lanes), various traffic light timings (from 60 seconds to 120 second periods), and various rates of pedestrians flowing into the street. In Table 6.4, easy to pass means that nodes are willing to pass whenever there is room to pass, whereas hard to pass means that nodes are not willing to pass when they catch up other nodes even if there is enough room to pass in the right or left lane.

When nodes pass through the measuring point shown in Figure 6.1 and 6.2, the information of passing time, current speed, desired speed, node ID, and direction

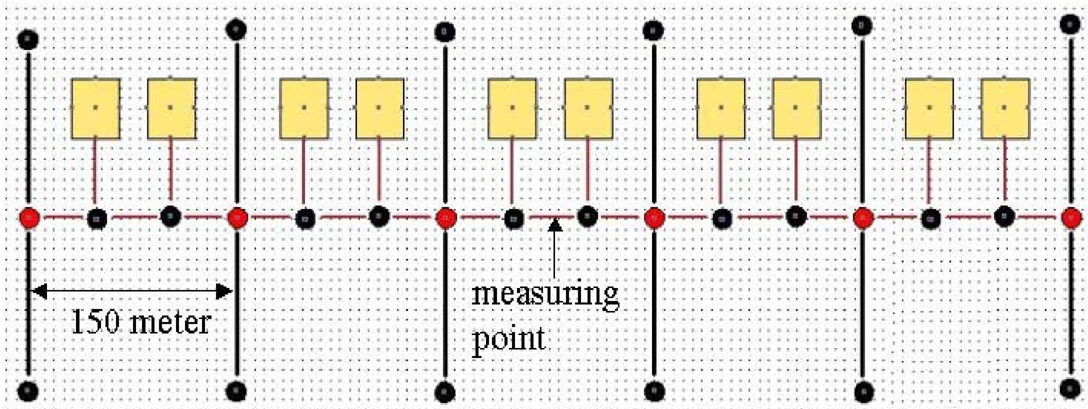

Figure 6.1: Map1 with buildings on measuring walkway. Red circle has a traffic light

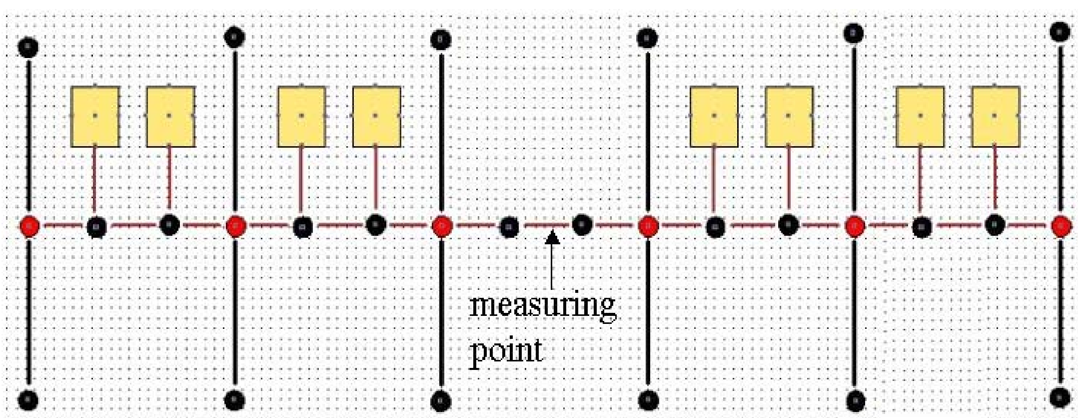

Figure 6.2: Map2 without buildings on measuring walkway

Table 6.4: Parameters for validation of pedestrian mobility

Number of pedestrian nodes	10,000
Simulation time	1,800 seconds
Maps	Map1, Map2
Number of lanes on walkway	4, 8, 16, 32
Traffic light period	60s, 90s, 120s
Passing rule	easy to pass, hard to pass, default mobility passing rule (Eq. 4.5)
Exit rate into the street	18,000s, 12,000s, 6,000s, 4,800s, 3,000s, 1,200s, 600s, 300s
Mobility model	constrained random waypoint, UDel mobility model

is recorded to analyze the burstiness of pedestrians.

Chapter 7

RESULTS

7.1 Traffic Light

Many people are congregated at the traffic light, increasing degree distribution and providing good connectivity. Figure 7.1 and 7.2 show that people are waiting when the traffic light is red and people move through the crosswalk after traffic light turns green. As you see, the appearance of moving through crosswalk looks similar with real world.

7.2 Platooning

Watching the movement of students on the University of Delaware campus, it is obvious that students make platooning at some parts of the area. Figure 7.3 shows that UDel mobility model properly generates platooning, but Figure 7.4 does not. This platooning will affect MANET connectivity and topology differently than the constrained random waypoint model does. We can imagine that unlike random waypoint model, node densities are not uniform, but nodes occur in clusters called platoons. Future algorithms such as cooperative relaying algorithms [59], [60] make use of clusters of nodes to form virtual antenna arrays and to enhance packet delivery.

7.3 Paddington City

Figure 7.5 and 7.6 show that some people are in offices and other people are moving to go to restaurants, stores, or residences at the time the snap shot shows. Since the UDel mobility simulator is designed to be scalable, the simulator

Figure 7.1: Snap shot of traffic light simulation when red light is on.

Figure 7.2: Snap shot of traffic light simulation when green light is on.

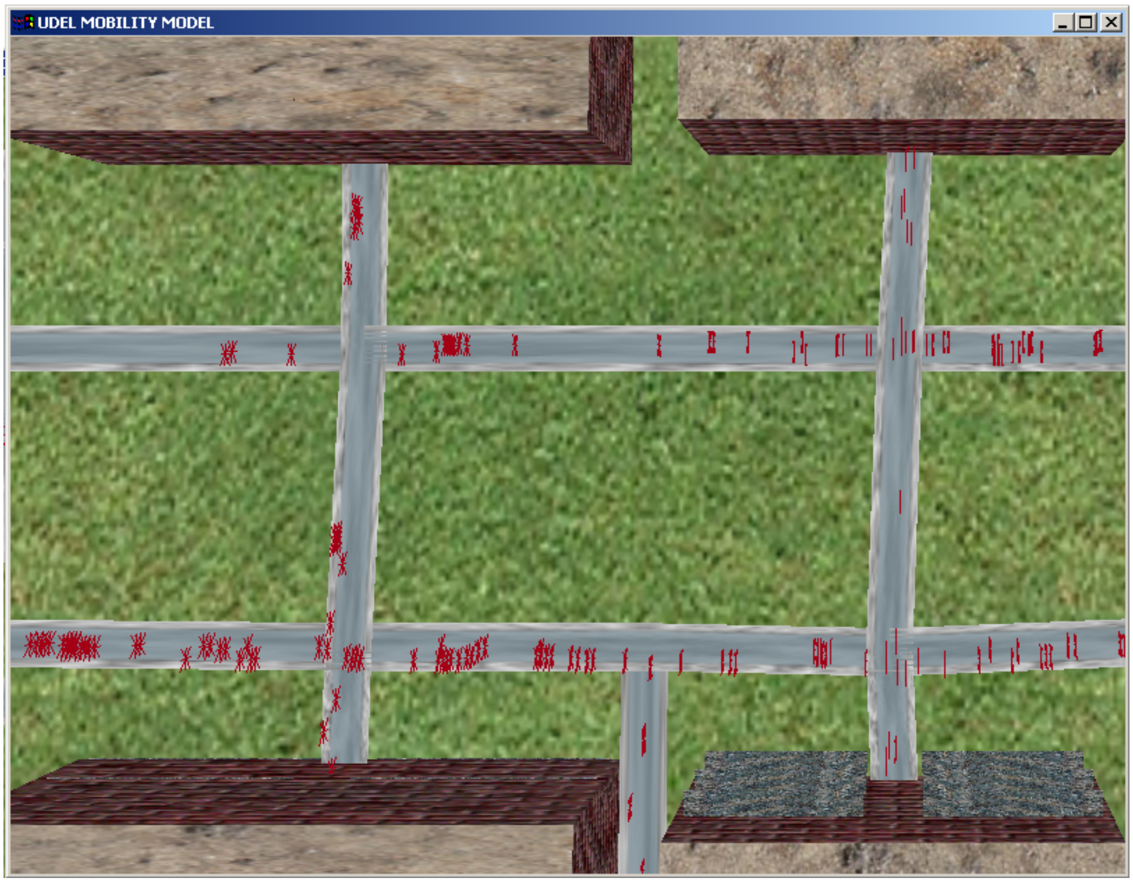

Figure 7.3: Snap shot of platooning simulation with UDel mobility model.

Figure 7.4: Snap shot of platooning simulation with constrained random waypoint model

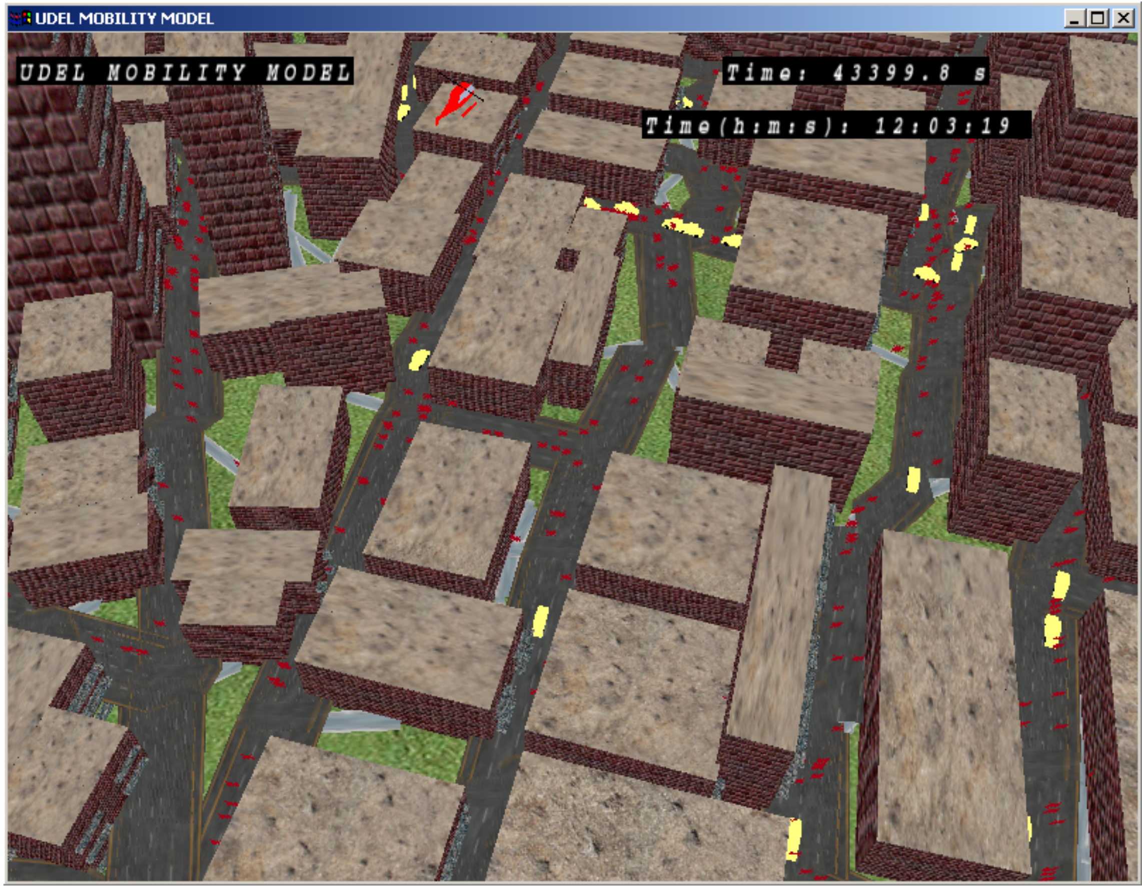

Figure 7.5: Snap shot of Paddington city simulation with normal view

can accommodate a huge number of mobile nodes in a big city. In this simulation, 10,102 mobile nodes are used.

7.4 Validation of Pedestrian Mobility

The burstiness of pedestrians, or in the terminology of traffic engineering, pedestrian platoons, has been investigated by Pushkarev and Zupan [36]. Their work has served as the basis for the pedestrian traffic engineering guidelines set forth in the Highway Capacity Manual [37]. The metrics of burstiness for pedestrian platoons are different from the ones typically used in studying data networks. Specifically,

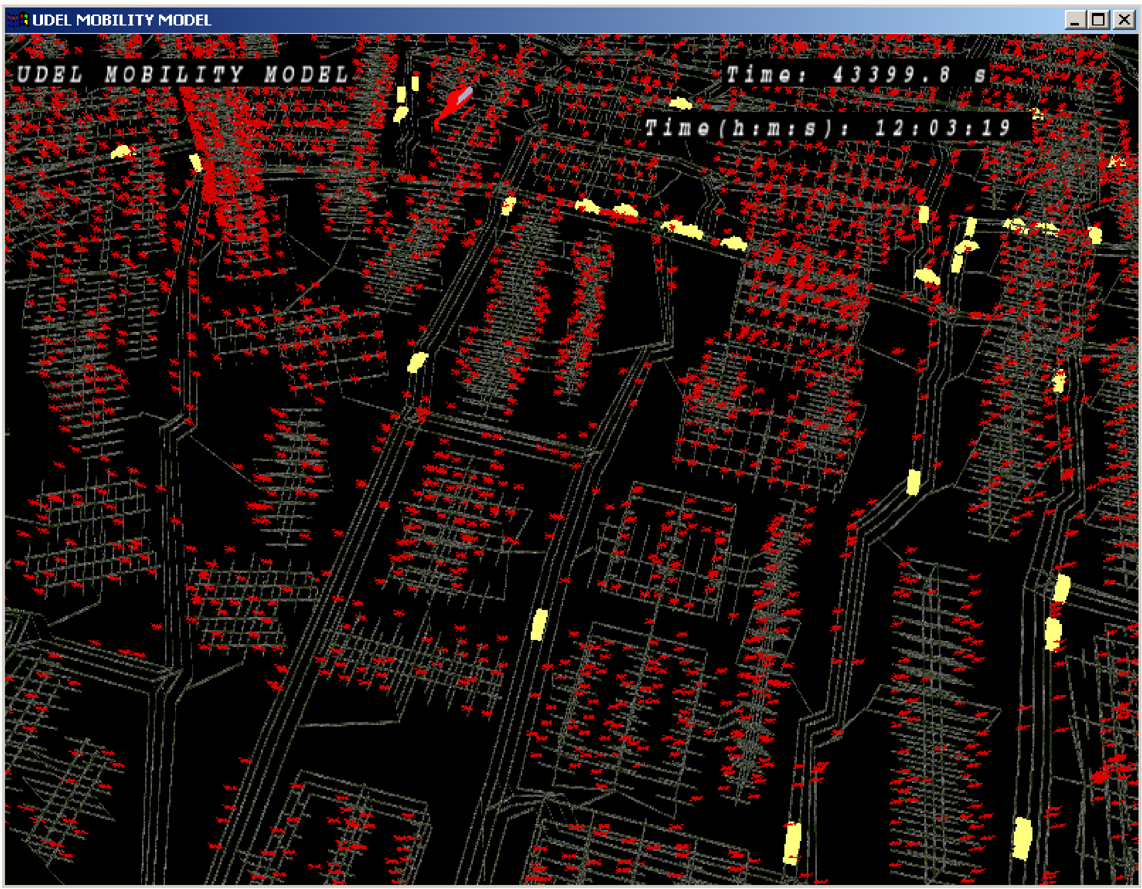

Figure 7.6: Snap shot of Paddington city simulation with inner view.

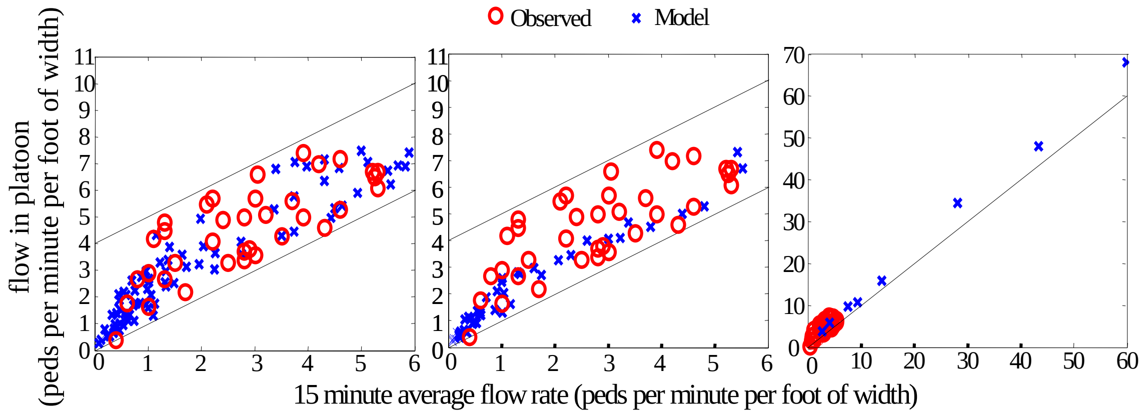

Figure 7.7: 15 Minute Average Flow Rate versus Flow Rate in a Platoon. The flow rate is the number of pedestrians that pass by the measurement point per minute divided by the width (in feet) of the sidewalk. The black line is the area of realistic values found by Pushkarev and Zupan.

Pushkarev and Zupan compare two flow metrics, the 15 minute average flow rate (AFR) and the flow rate during a platoon (PFR). A node is in a platoon if the local density of nodes exceeds the average density. As is shown in the Figure 7.7, the PFR is higher than the AFR. According to Pushkarev and Zupan, the larger the PFR compared to the AFR, the more bursty the pedestrian traffic is. The study of Pushkarev and Zupan was not focused on finding the frequency of specific flow rates, but to examine what combinations of AFR and PFR occur on urban sidewalks. Thus, we use this data as a baseline with which we compare the pedestrian mobility model described in Chapter 4.

As can be seen from the left-hand plot in Figure 7.7, the mobility model described above generates combinations of PFR and AFR that are realistic. The center plot in Figure 7.7 shows the data set collected by Pushkarev and Zupan and a set of data generated by the mobility model but where nodes pass whenever there is room to pass, i.e., P (desire to change lanes) = 1 as opposed to what is given in Eq. 4.5. Clearly, increasing the propensity to change lanes acts to decrease the burstiness so that some realistic levels of burstiness never occur. Finally, the right-hand plot in Figure 14 shows Pushkarev and Zupan's data compared to data generated by the mobility model but where there are no inter-pedestrians dynamics, i.e., nodes move along lane irrespective of other nodes. Such mobility allows, for example, nodes to exceed the distance-speed relationship. As shown in Figure 4.5, ignoring inter-node dynamics results in unrealistic levels of congestion (according to Pushkarev and Zupan the flow rate rarely exceeds 18).

Chapter 8

CONCLUSION AND FUTURE WORK

8.1 Conclusion

This thesis presented a realistic mobility model and methodology for use of simulating urban mesh networks. The model is based on extensive data, and hence the model and parameters are derived from what has been observed. The model is composed of three sub-models, the activity model, the task model, and the agent model. The activity model is based on a recent US Labor Department study of time-use in America, that collected detailed information about how people spend their time. The task model focuses on the mobility of people inside office buildings. This model is based on work in the area of meetings analysis [38]. The agent model determines how mobile nodes interact with each other as they move toward their destinations. This model is based on the work of Pushkarev and Zupan [36]. The technique of discrete event simulation is used to imitate realistic movement of mobile nodes. By carefully investigating necessary discrete events and applying derived mobility models to simulation at the appropriate time, the UDel mobility simulator correctly reflects the behavior of mobile nodes.

Currently, eight papers ([6], [41], [33], [34], [61], [59], [60], [62]) have been based on UDel mobility model. In the near future, the UDel mobility simulator will be hoped to help many research projects, especially for those who evaluate network performance in an urban mesh network.

8.2 Future Work

While this thesis has showed several aspects of realistic mobility models, much research still remains to be done in UDel mobility modeling. The following roughly sums up the future research.

For car models,

- Parking mobility model
- Delivery truck mobility models

For pedestrian models,

- Subway and train mobility models
- Non-office worker mobility models
- Morning and evening mobility models
- Disaster mobility models

It is necessary to include parking mobility models in the simulator. People take their cars and park at parking lots including street parking and do their own work. Another aspect in the city is that many delivery trucks pass through the modeled area, causing much traffic every day. Thus, we also need to consider this case.

Subway and train mobility model is important if we consider realistic mobility models because many people will go to work by subway or train, and few people travel entirely by foot in a city. UDel mobility model mainly focuses on the work day and office workers. To better evaluate performance of mesh networks, we should not ignore non-office workers activities and morning and evening activities. If we do that, we can simulate behavior of mobile nodes for 24 hours like real life. The other important issue is the disaster mobility model. The difference between normal and crisis situations will result in different performance of MANET protocols. Unlike the normal situation, it is difficult to produce experiments of crisis because few of us ever experience disasters often enough to develop an intuition of the mobility during a disaster. However, as one way, if we survey the extensive body of disaster and crisis research available at the University of Delaware's Disaster Research Center (DRC), disaster models could be developed.

BIBLIOGRAPHY

[1] A. S. Tanenbaum, *Computer Networks.* Prentice Hall PTR, 2003.

[2] S. Corson and J. Macker, "RFC 2501: Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations.," 1999.

[3] R. Poor and B. Hodges, "Reliable wireless networks for industrial systems." Ember Corporation, 2002.

[4] Philadephia Mayor's Office, "Government leaders teleconference - wireless: The 21st century technology," 2004.

[5] P. Ball, "Introduction to discrete event simulation," *DYCOMANS workshop*, pp. 367–376, May 1996.

[6] S. Bohacek and V. Sridhara, "The graph properties of MANETs in urban environments," *42nd Annual Allerton Conference*, 2005.

[7] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communication and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking:Research, Trends and Applications*, vol. 2, no. 5, pp. 483–502, 2002.

[8] F. Bai, N. Sadagopan, and A. Helmy, "IMPORTANT- a framework to systematically analyze the impact of mobility on performance of RouTing protocols for adhoc NeTworks," *IEEE INFOCOM 2003*, 2003.

[9] A. Jardosh, E. M. BeldingRoyer, K. C. Almeroth, and S. Suri, "Towards realistic mobility models for mobile ad hoc networks," *MobiCom 2003*, 2003.

[10] J. Tian, J. Hahner, C. Becker, I. Stepanov, and K. Rothermel, "Graph-based mobility model for mobile ad hoc network simulation," *IEEE 35th Annual Simulation Symposium*, pp. 337–344, April 2002.

[11] S. Ray, "Realistic mobility for MANET simulation," Master's thesis, The University of British Columbia, 2003.

[12] M. Schreckenberg and S. D. Sharma, *Pedestrian and Evacuation Dynamics*. Springer Verlag, 2002.

[13] J. Fruin, *Pedestrian Planning and Design*. Metropolitan Association of Urban Designer and Evironmental Planners, 1971.

[14] D. Helbing, *Verkehrsdynamik: Neue Physikalische Modellierungskonzepte*. Springer Verlag, Berlin, 1997.

[15] F. Navin and R. J. Weeler, "Pedestrian flow characteristics," *Traffic Engineering*, 1969.

[16] L. F. Henderson, "On the fluid mechanics of human crowd motion," *Transportation Research*, vol. 8, pp. 509–515, 1974.

[17] A. Hanish and T. Schulze, "Online simulation of pedestrian flow in pubilc buildings," *Proceedings of the 2003 Winter Simulation Conference*, 2003.

[18] M. Florian, M. Mahut, and N. Tremblay, "A hybrid optimization-mesoscopic simulation traffic assignment model," *Proceedings of the 2001 IEEE Intelligent Transport Systems Conference*, 2001.

[19] V. J. Blue and J. L. Adler, "Cellular automata microsimulation of bi-directional pedestrian flows," *Journal of the Transportation Research Board*, vol. 1678, pp. 135–141, 2000.

[20] A. Schadschneider, "Bionics-inspired cellular automaton model for pedestrian dynamics," *Traffic and Granular Flow*, 2002.

[21] J. Dijkstra, A. J. Jessurun, and H. J. P. Timmermans, "A multi-agent cellular automata model of pedestrian movement," *Pedstrian and Evacuation Dynamics*, pp. 173–181, 2001.

[22] T. Schelhorn, D. O'Sullivan, M. Haklay, and M. Thurstain-Goodwin, "STREETS: An agent-based pedestrian model," Tech. Rep. 9, Center for advanced spatial analysis working paper, 1999.

[23] B. Jiang, "Multi-agent simulations for pedestrian crowds," *European Simulation Symposium and Exhibition*, pp. 383–387, 1998.

[24] K. Teknomo, Y. Takeyama, and H. Inamura, "Microscopic pedestrian simulation model to evaluate "lane-like segregation" of pedestrian crossing," *Proceedings of Infrastructure Planning Conference*, 2001.

[25] C. R. Rindt, J. E. Marca, and M. G. McNally, "Toward dynamic, longitudinal, agent-based microsimulation models of human activity in urban settings," *TRB 2003 Annual Meeting*, 2003.

[26] F. Tecchia, C. Loscos, R. Conroy, and Y. Chryanthou, "Agent behavior simulator (ABS): A platform for urban behavior development," *Proceedings of Game Technology*, 2001.

[27] K. Ashida, S.-J. Lee, J. Allbeck, H. Sun, N. Badler, and D. Metaxas, "Pedestrians: Creating agent bebaviors through statistical analysis of observation data," *Computer Animation*, 2001.

[28] S. P. Hoogendoorn and P. H. L. Bovy, "Pedestrian route-choice and activity scheduling theory and models," *Transportation Research Board 2003 Annual Meeting*, 2003.

[29] A. F. Abdelghany and H. S. Mahmassani, "Temporal-spatial micro-assignment and sequencing of travel demand with Activity/Trips chains," *Transportation Research Board 2003 Annual Meeting*, 2003.

[30] R. J. Beckman, "The transportation analysis simulation system (TRANSIMS) - the dallas-ft. worth case study," Tech. Rep. LAUR-97-4502LANL, Lost Alamos National Laboratory, 1997.

[31] S. P. Hoogendoorn, M. Hauser, and N. Rodrigues, "Application of microscopic pedestrian flow simulation to station design evaluation in lisbon train stations," *TRB 2004 Annual Meeting*, 2004.

[32] Scalable Network Technologies, "The QualNet simulator." Tech. Rep. http://www.qualnet.com/.

[33] J. Kim and S. Bohacek, "A survey-based mobility model of people for simulation of urban mesh networks," *MeshNets 2005*, 2005.

[34] V. Sridhara, J. Kim, and S. Bohacek, "Models and methodologies for simulating urban mesh networks," *IEEE MobiWac 2005*, 2005.

[35] S. Bohacek and V. Sridhara, "The UDel models - MANET mobility and path loss in an Urban/Suburban environment," *Submitted*, 2004.

[36] B. Pushkarev and J. M. Zupan, *Urban Space for Pedestrians.* MIT press, 1975.

[37] Transportation Research Board, *2000 Highway Capacity Manual.* Washington, D.C.: National Research Council, 2000.

[38] N. C. Romano and J. F. Numamaker, "Meeting analysis: Findings from research and practice," *Proceedings of Teh 34th Hawaii International Conference on Systems Science*, 2001.

[39] A. Szalai, *The Use of Time.* The hague: mouton, 1972.

[40] U. D. of Labor Bureau of Labor Statistics, "American time use survey (ATUS)." http://www.bls.gov/tus/, 2003.

[41] V. Sridhara, J. Kim, and S. Bohacek, "Performance of urban mesh networks," *Submitted*, 2005.

[42] FHWA, *CORSIM User's Manual, Version 5.0.* ITS Research Division, FHWA, 2000.

[43] J. E. Hummer, "Unconventional left-turn alternatives for urban and suburban arterials," *ITE Journal*, vol. 68, 1998.

[44] A. Kamarajugadda and B. Park, "Stochastic traffic signal timing optimization." Tech. Rep. UVACTS-15-0-44, Center for transportation studies at the university of Virginia, 2003.

[45] R. R. Panko and S. T. Kinney, "Meeting profiles: Size, duration, and location," *Proceedings of The 28th Annual Hawaii International Conference on Systems Science*, 1995.

[46] R. K. Mosvick and R. B. Nelson, *We've Got to Start Meetings Like This!* Glenview, Illinois: Scott, Foresman and company, 1987.

[47] S. J. Older, "Movement of pedestrian on footways in shopping street," *traffic engineering and control*, pp. 160—163, 1968.

[48] F. P. D. Navin and R. J. Wheeler, "Pedestrian flow characteristics," *traffic engineering*, pp. 30—36, 1969.

[49] D. Oeding, "Traffic loads and dimensions of walkways and other pedestrian circulation facilities," *Strassenbau and strassenverkehrstechnik*, vol. 22, 1963.

[50] S. Shekleton, "A GPS study of car following theory," *Conference of Australian Institutes of Transport Research (CAITR)*, 2002.

[51] T. Dijker, P. H. L. Bovy, and R. G. M. M. Vermijs, "Car following behavior in different flow regimes," *Motorway Traffic Flow Analysis*, pp. 49–70, 1998.

[52] J. Piao and M. McDonald, "Analysis of stop and go driving behavior through a floating vehicle approach," *Proceedings Of the IEEE Intelligent Vehicles Symposium*, 2003.

[53] D. Helbing, "Sexual differences in human crowd motion," *Nature*, vol. 240, p. 252, 1972.

[54] D. Helbing, "The statistics of crowd fluids," *Nature*, vol. 229, p. 381, 1971.

[55] G. K. Still, *Crowd Dynamics*. PhD thesis, University of Warwick, 2000.

[56] J. Du and L. Aultman-Hall, "An investigation of the distribution of driving speeds using in-vehicle GPS data," *Vermont Institute of Transportation Engineers Annual Meeting*, 2004.

[57] Y. Zhang, L. E. Owen, and J. E. Clark, "A multi-regime approach for microscopic traffic simulation," *The 77th TRB Annual Meeting*, 1998.

[58] K. I. Ahmed, *Modeling Drivers' Acceleration and Lane Changing Behavior*. PhD thesis, MIT, 1999.

[59] J. Kim and S. Bohacek, "Multi-hop cooperative relaying for mobile netoworks," *Submitted*, 2005.

[60] J. Kim and S. Bohacek, "Selection metrics for multihop cooperative relaying," *MedHocNet 2005*, 2005.

[61] S. Bohacek, A. Ilic, and V. Sridhara, "On the predictability of link lifetimes," *WiOpt 2005*, 2005.

[62] J. Kim and S. Bohacek, "Enhancing routing performance by exploiting diversity," *Submitted*, 2005.